

Elementos de experiencia de usuario para sistemas de detección de plagio.

Enma Lidia Muñoz García, Abel Meneses Abad, Alberto Barrón Cedeño

Abstract

Se presentan los resultados del estudio de 13 sistemas de detección de plagio [PDS] atendiendo a sus elementos de experiencia de usuario [UX], algunos de ellos reconocidos por artículos científicos como los más utilizados o eficientes dentro de esta gama de aplicaciones. Se hacen referencia de forma general a otros PDS estudiados menos importantes. Se elabora una propuesta de los elementos más importantes a utilizar en el módulo de detección de plagio de la plataforma cubana Sunshine, un repositorio institucional avanzado de acceso abierto.

Introducción

El diseño de la experiencia de usuario consiste entender y cubrir todas las acciones del usuario de manera que se satisfagan sus expectativas en todo el proceso. Cuando un usuario enfrenta un sitio web bien diseñado no necesita manual para encontrar lo que necesita u orientación para saber donde se encuentra.

Los sistemas de detección automática de plagio en documentos, son un tipo de sistema informático de alta complejidad que apenas tienen una década de desarrollo intensivo.[29] La necesidad de analizar extensos documentos vs enormes fuentes de información como internet, ha sido un reto tanto para la algoritmia computacional, como para la UX. En tal sentido debe valorarse que, mientras las ciencias de la computación deben interpretar el lenguaje natural propio del texto, modelar fenómenos complejos como la paráfrasis y extraer fuentes posibles desde cualquier página web en la Internet o basándose en el estilo del autor[6], en un PDS el usuario solo desea que de forma sencilla se le presente sí el texto es plagiado o no, y en cuáles partes.

Sunsine es un repositorio institucional avanzado de acceso abierto, denominado de esta forma por sus cercanías en funcionalidades a sistemas de recomendación, y por implementar servicios complementarios muy importantes para una institución educativa como: la detección de plagio. Esta plataforma contempla en uno de sus módulos la elaboración de un sistema capaz de detectar los diferentes tipos de texto-reusado, entre ellos los que infringen el derecho de autor.¹

¹Los autores del presente artículo son miembros del equipo de desarrollo de este proyecto.

Análisis de UX de Sistemas Internacionales de DP.

Como estrategia fundamental para el diseño de la experiencia de usuario para este sistema de detección de plagio se realizó un estudio de homólogos de los PDS más reconocidos a nivel internacional. Estos sistemas se clasificaron de acuerdo al origen de las fuentes que recupera y en las que basa su análisis (ficheros en bases de datos internas al sistema, o en redes externas como internet): **hermética, web o ambos**. Kakkonen y Mozgovoy en su estudio "Hermetic and Web Plagiarism Detection Systems for Student Essays— An Evaluation of The State-Of-The-Art" coinciden con esta clasificación. En este estudio se mencionan como los sistemas de detección hermética de mayor relevancia: **WCOPYFIND**, **Sherlock** y **AntiPlagiarist**[17]. Estos sistemas en general le permiten al usuario realizar un análisis de plagio, proponiendo además opciones de configuración para una búsqueda más refinada. Por ser Sunshine una aplicación web, para este estudio de homólogos se desea proponer un diseño visual que mejore las aplicaciones desktop anteriores. Vale mencionar también que en cuestiones del negocio se considera a los sistemas web de mayor alcance para detectar plagio, pues pueden ser accedidos con mayor facilidad, posibilitando la retroalimentación de su repositorio central de documentos con nuevos ejemplos de los usuarios que utilizan el sistema, retro-alimentando a su vez a todos los usuarios con los nuevos cambios.

Los **sistemas de detección de plagio automáticos** han crecido en número y calidad, la composición de este grupo sin embargo es extraordinariamente cambiante pues, siendo los sistemas de detección automática de plagio un servicio para instituciones que gestionan publicaciones, revistas o patrimonio documental (como las universidades o bibliotecas públicas), algunos sistemas se vuelven comerciales y otros proyectos al necesitar más recursos son engavetados. Este estudio basa su análisis en los principales ejemplos certificados en la actualidad por su uso o posibilidades técnicas:

1. **Turnitin:** Es una solución integral para evaluar trabajos escritos. Los instructores pueden superponer o alternar entre los informes de OriginalityCheck, GradeMark y PeerMark para apreciar y comprender el trabajo escrito en su totalidad.[12] Las principales características de estos tres productos son las siguientes:
 - a) OriginalityCheck: ayuda a los instructores a revisar el trabajo de los estudiantes y a detectar citas incorrectas o posibilidades de plagio al comparar el trabajo con la base de datos de comparación de texto más exacta y completa a nivel mundial. Entre sus funcionalidades presenta:
 - 1) Muestra las partes del trabajo del estudiante que coinciden con nuestra base de datos para que los instructores puedan comprender rápidamente cuánto del trabajo no es original.
 - 2) Las fuentes coincidentes aparecen en el trabajo en un formato fácil de entender, revelando las fuentes codificadas por color correspondientes al trabajo no original.

- 3) Controla el tipo de información que aparece en el informe de originalidad, filtrando material bibliográfico, citas, y coincidencias menores.[13](Véase la figura 16 en los Anexos)
- b) Grademark: ahorra tiempo a los instructores y le proporciona comentarios constructivos y pertinentes a los estudiantes habilitando comentarios editoriales, personalizados y de QuickMark directamente sobre sus trabajos. Entre sus funcionalidades presenta:
 - 1) Permite ajustar los criterios, valores y escalas de sus matrices de evaluación para garantizar la claridad en la entrega de comentarios a los estudiantes.
 - 2) Permite analizar e identificar las áreas de mayor dificultad, para hacer un seguimiento del progreso de los estudiantes con estadísticas y gráficos completos.
 - 3) GradeMark ayuda a los instructores a proveer comentarios más completos y relevantes acerca de sus trabajos, lo cual mejora la calidad de su escritura y trabajo académico, a través de la interacción en línea.[11](Véase la figura 17 en los Anexos)
- c) PeerMark: Los estudiantes aprenden de sus instructores, pero también de sus compañeros. Facilita la revisión por compañeros para que los estudiantes puedan evaluar sus trabajos entre sí y aprender el uno del otro. Entre sus funcionalidades presenta:
 - 1) Permite la distribución de los trabajos de los estudiantes electrónicamente y en cualquier momento, para que los estudiantes puedan revisar los trabajos de sus compañeros.
 - 2) Logra adaptar el proceso de revisión por compañeros al flujo de trabajo de su clase usando opciones como la revisión anónima por compañeros.
 - 3) Ayuda a los estudiantes a avanzar en sus trabajos y a hacer los comentarios apropiados a través de preguntas estándar o personalizadas.
 - 4) Desarrolla las habilidades de pensamiento crítico de los estudiantes y los anima a compartir y a ayudar a otros estudiantes en la clase.[14](Véase la figura 18 en los Anexos)
2. **EVE2 (Essay Verification Engine):** Sistema desarrollado por Canexus. Es un software privativo, con un costo de 30\$.² No mantiene una base de datos de sus propios ensayos o textos. Su método consiste en la búsqueda en internet de ensayos haciendo uso de motores de búsqueda existentes. El programa se instala en la propia computadora del usuario y le permite seleccionar a partir de tres tipos posibles de detección: rápido, medio y fuerte. Solo acepta hasta diez archivos a la vez. Cuando el sistema realiza la búsqueda de plagio muestra una vista con el porcentaje de plagio encontrado (Véase la figura 15 en los Anexos), dándole la opción al usuario de obtener otra vista con los resultados, tal como se observa en la siguiente figura: [17]

²<http://www.canexus.com>

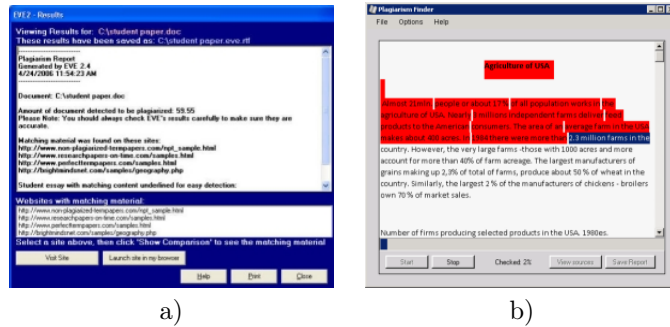


Figura 1: a) Resultados EVE2 [30]. b) Resultados Plagiarism-Finder[28]

3. **JPlag:** Es un sistema que encuentra similitudes entre múltiples conjuntos de archivos de texto. No se limita a comparar bytes de texto, sino que es consciente de la sintaxis del lenguaje de programación y la estructura del programa y por lo tanto es resistente contra muchos tipos de intentos de disfrazar similitudes entre archivos plagiados. JPlag actualmente soporta Java, C #, C, C++, y texto en lenguaje natural. Se suele utilizar para detectar la copia inadmisibles de los programas de clase de estudiantes de programación. También se puede utilizar para detectar partes de software robados entre grandes cantidades de texto fuente o módulos que han sido duplicados (y sólo ligeramente modificados). JPlag se ha utilizado con éxito por peritos en varios casos de análisis forense de propiedad intelectual. JPlag compara los programas, tratando de cubrir uno de los programas (preferiblemente grande) con secuencias del otro programa. [21]

| | |
|-------------------------------------|--|
| Search Results | |
| JPlag | |
| Directory: | I:\jplag\testdata\anonym |
| Programs: | 617012 - 703982 - 705604 - 720061 - 729057 - 742941 - 745586 - 748814 - 750734 - 769531 - 786143 - 788172 - 791299 - 791642 - 792145 - 793455 - 795417 - 808568 - 826366 - 826606 - 826764 - 826833 - 826877 - 826946 - 827052 - 827063 - 827132 - 827290 - 827654 - 827825 - 828099 - 829036 - 829503 - 829683 - 842771 - 861005 - 861061 - 861130 - 861196 - 861301 - 861641 (maxSimTest) - 861732 - 862224 - 862246 - 862326 - 862531 - 862564 - 863261 - 863307 - 863625 - 870711 - 878135 - 908172 - 942261 - 942909 - 943151 - 944063 - 944814 |
| Language: | Java1.7 Parser |
| Submissions: | 58 (1 has not been parsed successfully) |
| Invalid submissions (see log file): | 861221 |
| Matches displayed: | 25 (Threshold: 20.1%) (average similarity) 25 (Threshold: 22.0%) (maximum similarity) |
| Date: | 2013-04-24 |
| Minimum Match Length (sensitivity): | 9 |
| Suffixes: | .java, .jav, .JAVA, .JAV |

Figura 2: Reporte de resultados de JPlag[22]

4. **Plagiarism-Finder (PF; versión 1.3.0):** Este sistema trabaja similar

a EVE2. Se instala en la computadora del usuario y busca en internet las posibles ocurrencias de fragmentos de texto de la colección local de documentos. PF le permite al usuario ajustar el algoritmo de detección a través de dos parámetros: el registro de la longitud(en palabras) que se necesita chequear y el incremento(en palabras) que define la longitud del paso por donde se harán lo avances al siguiente registro (una secuencia de palabras) en el documento. (Véase el reporte resultados de PF en la figura20en los Anexos)

5. **CoReMo:** Es un excelente sistema desarrollado en España, cuyos algoritmos han demostrado estar en la lista de las técnicas actuales más eficientes para enfrentar los problemas computacionales actuales de la detección de plagio.[31] Aún no disponible su interfaz web en internet, es según sus propios autores un sistema pensado para procesar paralelo grandes volúmenes de datos. Utiliza un sistema de alta precisión de recuperación de información, la técnica de n-grams y un módulo solo para el alineamiento de textos. Puede detectar a través de estas técnicas con un recall elevado casos de plagio croslingual. En estos momentos implementa su interfaz de usuario web, y adiciona soporte para lenguas raras como el Hindi.[32][33]
6. **SafeAssign:** Es un servicio de prevención de plagio. Este servicio ayuda a los educadores a evitar el plagio mediante la detección de contenidos no originales en los documentos de los estudiantes. Además de actuar como elemento de disuasión de plagio, también tiene características diseñadas para ayudar a educar a los estudiantes sobre el plagio y la importancia de la debida atribución de cualquier contenido prestado. Para realizar el proceso de coincidencias SafeAssign chequea todos los documentos presentados en contra de las bases de datos siguientes:

- *Internet:* índice completo de los documentos disponibles para el acceso público a través de Internet.
- *ProQuest ABI/Inform database* con más de 1.100 títulos de publicaciones y alrededor de 2,6 millones de artículos desde los años 90 hasta la actualidad, actualiza semanalmente (acceso exclusivo)
- *Archivos institucionales de documentos* que contienen todos los documentos presentados a SafeAssign por los usuarios en sus respectivas instituciones.
- *Base de datos de referencia global* que contiene los documentos que fueron ofrecidos voluntariamente por los estudiantes de las instituciones clientes de Blackboard para ayudar a prevenir la contaminación cruzada institucional plagio.[3]

Un informe de originalidad SafeAssign resalta los bloques de texto en los documentos presentados que responden a las fuentes de referencia y enlaces a los documentos encontrados en Internet o en bases de datos de contenido permitidos. También muestran que los índices de similitud para cada oración y permite ver a los instructores una comparación línea por línea de texto potencialmente poco original de los documentos presentados

y los documentos externos que coincidan.[19] A continuación se muestra una imagen de dicho informe:

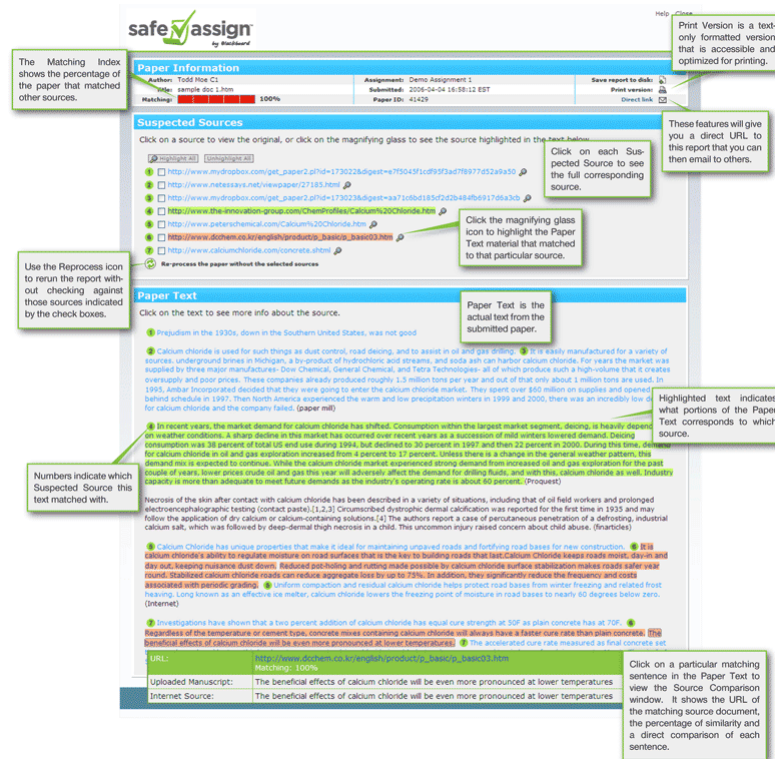


Figura 3: Reporte de originalidad SafeAssign [23]

7. **CrossCheck (2009):** Es el sistema oficial que utilizan instituciones como la IEEE para la revisión automática de texto reusado en las publicaciones científicas que archivan. Es muy profesional tanto en sus algoritmos como a nivel de interfaz. Esta plataforma permite una rápida manipulación de los ficheros contenidos por ella organizados a través de carpetas similar a como realizan su gestión los exploradores de directorio, contiene múltiples funcionalidades relacionadas con el tratamiento de documentos: varios modos de visualización, excluir resultados de cadenas cortas o excluir determinadas coincidencias; adicionalmente permite la gestión de usuarios.[15]

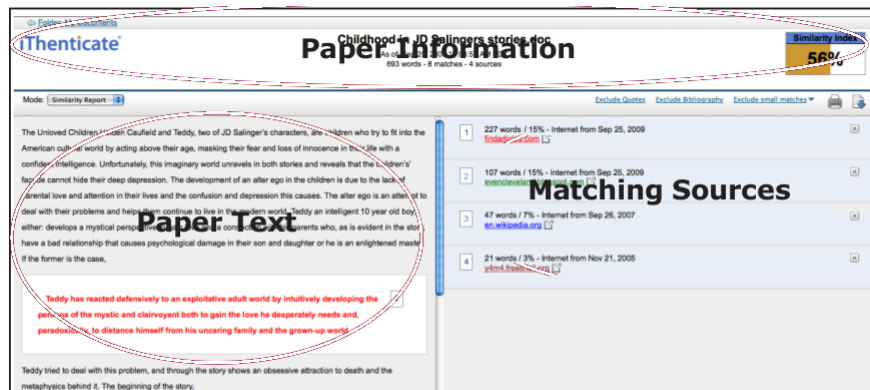


Figura 4: Screenshot del Reporte de originalidad de la plataforma CrossCheck [15]

8. **CitePlag 2013:** Es un sistema de detección de plagio basado en una técnica ideada por Bela Gipp, que analiza la similitud de las citas contenidas en los documentos. Este sistema ha sido liberado en el 2013 y posee una interfaz web muy profesional que puede ser evaluada en su sitio oficial, CitePlag. Posee numerosas funcionalidades visuales como la revisión de resultados previos, el análisis utilizando diferentes métricas y un intuitivo reporte de similitud.[8][6]

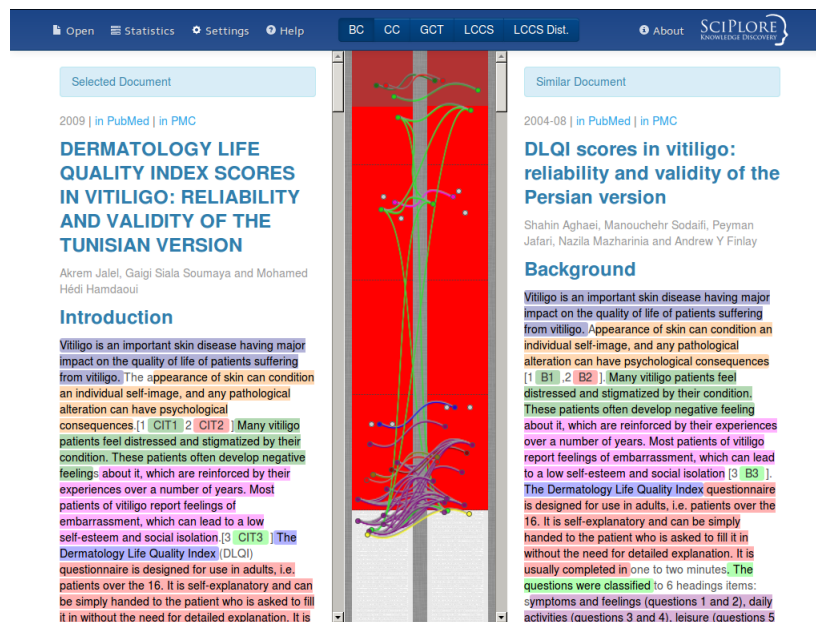


Figura 5: Screenshot Ejemplo Reporte de Originalidad CitePlag [34]

9. **Plagiarisma.net:** Este sitio permite buscar pequeños textos duplicados sin necesidad de usuarios. (Véase figura 19 en los Anexos) Tiene una fuerte política de seguridad para atender a sus usuarios, pero permite el acceso libre. Es bastante rápido obteniendo resultados en menos de un minuto. Devuelve las fuentes por cada oración que compone al texto. Utiliza Google, Yahoo y Babylon como fuentes de datos para encontrar las semejanzas. Permite analizar múltiples formatos de documentos. No presenta un reporte personalizado o con gran experiencia de usuario, este servicio ha sido diseñado para el uso común. Tiene búsqueda difusa y utiliza la distancia Levenshtein. Posee un servicio para permitir al usuario reescribir textos haciendo paráfrasis a partir de sinónimos, pudiendo definir el % de reescritura deseada.³

| Results | Query | Domains (cached links) |
|-------------------------|--|--|
| Unique | The University of Chicago Press manages the standards and rules for Chicago Style | |
| Unique | The principal handbook dedicated to Chicago Style is called "The Chicago Manual of Style," which you may see shortened to CMS or CMOS | |
| Unique | The University of Chicago Press produced the original Chicago Style Manual in 1906 | |
| Err | The Chicago Style Manual offers guidance and criteria for every parts of writing; it is not confined to producing formal and academic papers | |
| Err | This handbook offers pointers and rules on English grammar, proper use of acronyms, and correct punctuation | |
| Err | Besides the printed version, you can visit the organization's website www | |
| 2 results | chicagomanualofstyle.org for additional information | |
| Unique | Chicago Style also has a second handbook called "A Manual for Writers of Term Papers, Theses, and Dissertations," written by Kate <small>Did you mean: "Chicago Style also has a second <i>hand book</i> called <i>Basics</i>: A Manual for Writers of Term Papers, Theses, and Dissertations, <i>Basics</i>, written by Kate."</small> | |
| About 4,120,000 results | Turabian | press.uchicago.edu press.uchicago.edu en.wikipedia.org en.wikipedia.org turabian lib.usa.edu www2.liu.edu |
| Unique | You will often hear that Turabian Style and Chicago Style are one in the same style because they have identical but slightly refined rules | |

Figura 6: Informe de resultados de Plagiarisma.net[35]

10. **Viper:** Software libre para Windows que promueve la detección de plagio en ambos sentidos, es decir alienta el hábito de la revisión de plagio también de parte del que realiza el ensayo o artículo, no solo de quien lo revisa. De modo que resulta una vía factible en el ambiente educativo. De ahí que se diga que es la alternativa libre para Turnitin. Realiza una revisión que ofrece las siguientes funcionalidades:

- Escaneo de plagio literalmente contra miles de millones de documen-

³<http://plagiarisma.net>

tos, incluyendo libros, revistas y sitios web.

- Un informe con un código de colores que identifica las posibles áreas problemáticas, todas las referencias utilizadas en todo el documento destacadas en rojo de manera que se pueda comprobar que se han utilizado y citado correctamente.
- Permite ver exactamente que áreas del documento pueden ser vulnerables a las acusaciones de plagio de un profesor o tutor.

De manera general el reporte resume:

- Si el porcentaje de texto encontrado en las fuentes es apropiado para el documento subido y muestra dicho texto bien identificado.
- Si el porcentaje de citas es adecuada para los propósitos del documento subido.
- Si las coincidencias encontradas en el texto están correctamente referenciadas y en el estilo adecuado, según las normas de la escuela o universidad.
- Por-ciento de texto que se sospecha que haya sido plagiado.[37]

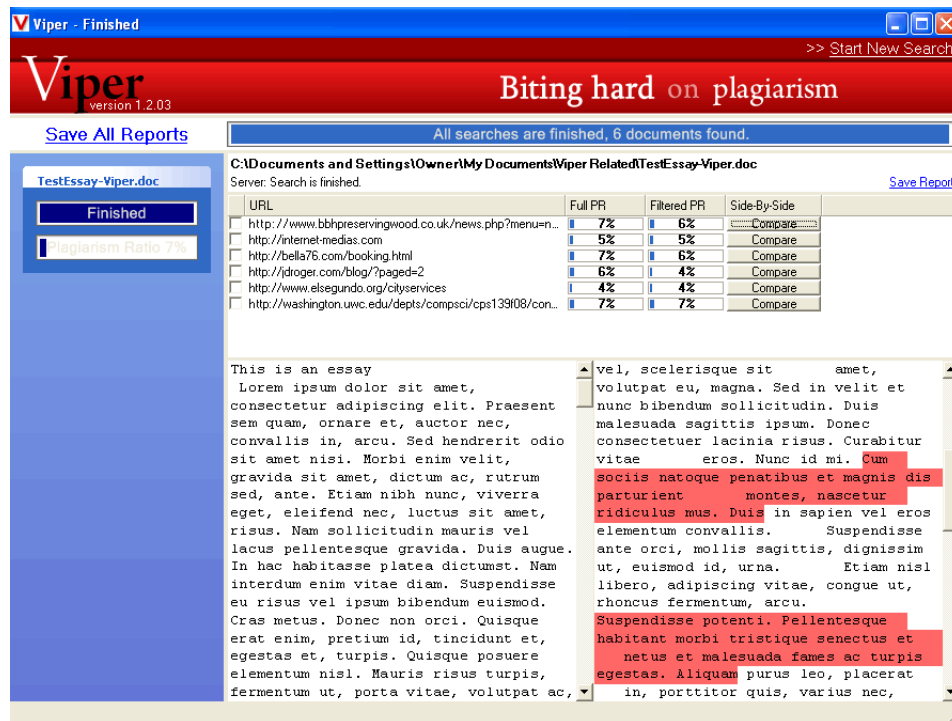


Figura 7: Reporte de Viper[1]

11. **SeeSources.com:** Es un servicio web que le permite al usuario tanto copiar/pegar como subir un archivo (en MS Word, HTML, o formato ASCII) con el propósito de detección de plagio. No se le ofrecen opciones al usuario. La búsqueda tiene lugar en dos fases: búsqueda de fuente y búsqueda profunda[17].
<http://www.flickr.com/photos/77913019@N00/5241614589/>
12. **Plagiarism-detect.com**
Este sitio de libre acceso actualmente presenta problemas al generar los reportes, los problemas son fundamentalmente técnicos y de difícil solución. Un mensaje en el sitio explica: “Probablemente no podremos seguir ofreciendo el servicio de detección de plagio. La razón fundamental es que este depende en gran medida de los proveedores internacionales de indexación de Internet. En estos momentos el API de Microsoft de ‘Bing Search’ y el de Yahoo no son libres. Además el API de Google ha sido despreciado, y es solo cuestión de tiempo para que Google lo deshabilite...”⁴
13. **Antiplag**
PDS Web utilizado en el repositorio central del Ministerio de Educación de la República de Eslovaquia. Este sistema cuenta con tres etapas, en las cuales transforma los texto en una estructura eficiente para su comparación con el resto, compara pasajes de texto identificando un umbral de palabras comunes, y finalmente elimina las superposiciones en la detección.[9] Es un sistema comercial soportado por SVOP Ltd., que procesa 80 mil tesis al año. Siendo comparable sus resultados a los del primer lugar en el PAN de 2011, detectando más del 70 % de los casos.[10] Su principal aval resulta su utilización masiva en la certificación de más de 200 mil tesis hasta mayo de 2012. [24]⁵

⁴<http://plagiarism-detect.com/>

⁵<http://www.svop.sk/en/antiplag.aspx>*

| Sistemas | Aspectos de funcionamiento | | | | | Componentes Visuales | | | |
|-----------------------|----------------------------|----------|--------|--------------------|------------|-----------------------------|---------------------|--------------------------|-----------------------|
| | App Desktop? | App Web? | FLOSS? | Analiza + 10 docs? | BD propia? | Compara el doc susp vs src? | ≡ links al doc src? | Señala Tipos de Plagio ? | Muestra %s de plagio? |
| Turnitin | X | O | X | X | O | O | O | X | O |
| EVE2 | O | X | X | X | X | X | O | X | O |
| Plagiarism Finder | O | X | X | O | X | X | O | X | O |
| JPlag | X | O | O | O | O | O | O | X | O |
| Coremo | X | O | X | O | O | — | — | — | — |
| SafeAssign | X | O | O | X | O | O | O | X | O |
| CrossCheck | X | O | X | X | O | O | O | X | O |
| CitePlag | X | O | O | O | O | O | X | X | X |
| Plagiarisma.net | O | X | X | X | X | X | O | X | O |
| Viper | O | X | O | O | O | O | O | X | O |
| SeeSources.com | X | O | O | X | X | X | O | X | O |
| Plagiarism-detect.com | X | O | O | X | X | — | — | — | — |
| SVOP Antiplag | O | X | O | X | X | ? | X | O | X |

Cuadro 1: Resumen de los resultados del estudio(1) y el estudio (2)

Otros sistemas que aparecen en la literatura mencionados en los artículos o en la web pero que no pudieron ser estudiados o cuyo análisis resulta difícil a partir de los datos encontrados son: PlaDeS, MyDropBox ahora perteneciente a la familia SafeAssign.

Experiencia de Usuario en Sistemas Web de Detección de Plagio.

Para la concepción de Sunshine, especialmente la parte de detección de plagio, se hace necesario el estudio de los aspectos más relevantes del tema en aras de disipar las dudas que surgen en un tema poco explorado como este. Por ello que para el análisis y diseño de la experiencia de usuario en cuestión se propone la utilización de las 5 etapas del modelo de Garret.[5]

- Strategy Plane
- Scope Plane
- Structure Plane

- Skeleton Plane
- Surface Plane

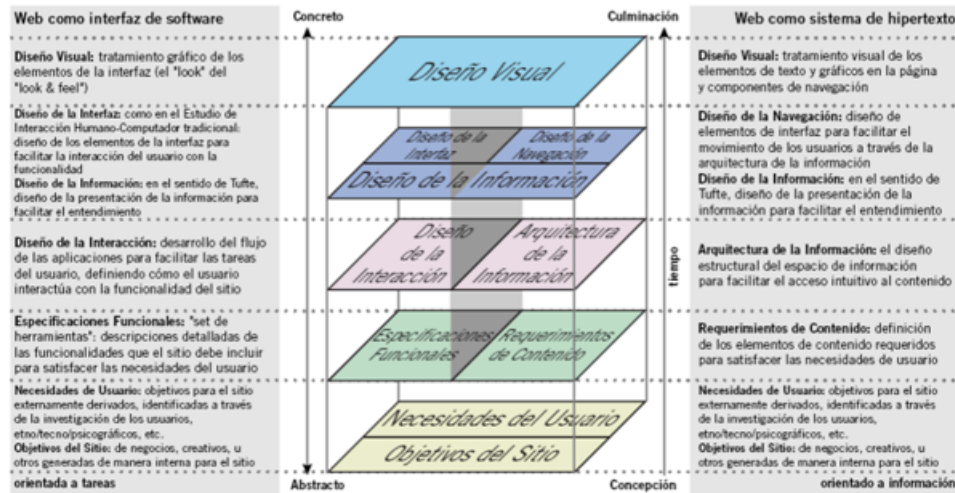


Figura 8: Elementos de la Experiencia de Usuario. Imagen original del libro *The Elements of User Experience*, Garret, 2002, traducción Javier Velazco.

Adicionalmente se realizaron estudios exploratorios de estos elementos de la *Experiencia de Usuario (UX)*⁶ a los sistemas de detección de plagio existentes a nivel mundial, descomponiéndose estos en elementos simples que podrían ser valorados en una solución final. A continuación se detalla el desarrollo de dichas etapas aplicadas al sistema de detección automático de plagio del proyecto *Sunshine*.

Necesidades del Usuario/ Objetivos del Sitio (Primer plano o Strategy Plane):

Esta etapa incorpora no solo la estrategia que los usuarios aportan al sitio, sino lo que esperan obtener de él.[5] En el caso de la detección de plagio los **objetivos** estratégicos a cumplir por el sitio serían los siguientes:

- Procesar los archivos subidos por el usuario en el menor tiempo posible.
- Utilizar los algoritmos óptimos para la detección de los distintos tipos de plagio.
- Mostrar en una interfaz amigable los resultados del análisis, de manera que los mismos sean entendibles para cualquier tipo de usuario.

⁶ *User Experience*

El otro aspecto a tener en cuenta aquí son las **necesidades de los usuarios**. Inicialmente Sunshine está concebido para ser un repositorio institucional de las universidades de Granma y Camagüey, esto define una audiencia con necesidades diversas. Es por ello que es necesario separarla en dos grandes grupos de usuarios: **novatos** y **avanzados**, donde los usuarios novatos representarían un tipo de "persona"⁷ con pocos conocimientos sobre la detección de plagio y cuyo interés será obtener el reporte de plagio de un documento; los usuarios avanzados serían los profesores o especialistas en el tema, los cuales además de obtener el informe de plagio, desearán colaborar en el mejoramiento de los algoritmos implementados, este tipo de sistema al poseer características para la retroalimentación colaborativa podría mantenerse en evolución gracias a dichos usuarios.[36] La recopilación de información sobre estos tipos de usuarios se ha realizado a través de la convivencia del equipo de desarrollo dentro de la propia institución.

Especificaciones Funcionales & Requerimientos de Contenido (Segundo plano o Scope Plane):

Todas las características o funcionalidades que poseerá el sitio están comprendidas en esta etapa. Tener bien definidos los requerimientos permitirá una distribución eficiente del trabajo.[5]

En la primera versión de la aplicación no fueron implementados todos los requerimiento iniciales, es por ello que para la segunda versión se volvieron a revisar los mismos, incluyendo todo lo referente a la parte de detección de plagio partiendo de la estrategia de la etapa anterior. Una vez que se realizó el estudio de homólogos quedó bastante claro lo que se podría desarrollar en el sitio, así que sirvió para refinar aun más los requisitos. A continuación se nombran algunos de los requisitos más relevantes a incorpora en esta versión de Sunshine:

- Administración de usuarios:
 - Registrar usuario, Autenticar usuario, Asignar rol manual o automático de acuerdo al comportamiento.
- Gestión de un área de trabajo personalizada al rol y por ende a sus permisos:
 - Gestionar “Mi perfil”, Gestionar “Mis Catálogos”, Gestionar “Reportes de Texto Reusado”.
- Procesamiento del Lenguaje Natural

⁷Técnica empleada en la experiencia de usuario para ayudar al equipo o al cliente a ponerse en los zapatos de los usuarios del producto final.[A Project Guide to UX Design- Russ Unger & Carolyn Chandler]

- Extracción de tokens, Identificación automática del idioma del documento, eliminación de palabras poco significativas según Lunh (Ponderación), Lematización de palabras, análisis de sinónimos, identificación de nombres propios, siglas y entidades, etiquetado de partes de la oración, etc.
- Recuperación de Información:
 - Posibilidad de indexar con varias estructuras de índice (Ej. tf-idf), Ordenar o jerarquizar los resultados, y análisis de datos procedentes del uso colectivo (collective intelligence).
- Servicios de análisis y recomendación:
 - Detección de plagio parafrástico, clasificación automática, recomendación de textos relacionados, detección de ediciones, detección y eliminación de duplicados.

Diseño de la Interacción y Arquitectura de Información (Tercer plano o Structure Plane):

Esta etapa define como los usuarios llegan a una página y hacia donde van a partir de ahí, además dónde se encuentran las categorías bajo las que se organiza la información. En el desarrollo de software tradicional la disciplina que tiene que ver con la creación de una experiencia de usuario estructurada es conocida como **diseño de interacción**.^[5]

Estructurar el desarrollo de contenido esta comprendido dentro de la **arquitectura de información**. Esta campo define un grupo de disciplinas que históricamente han tenido que ver con la organización, agrupamiento, ordenamiento y presentación del contenido. La Arquitectura de Información tiene que ver con como cognitivamente la gente procesa información, sus consideraciones son críticas en caso de productos orientados a la información (*information-oriented products*⁸) e incluso tener mas impacto en productos orientados a la funcionalidad (*functionality-oriented products*⁹)^[5]

En este caso, la idea es diseñar las pantallas que tienen que ver con la detección de plagio pensando en el comportamiento del usuario y cómo el sistema responderá en consecuencia. Para lograr un mayor entendimiento de como los componentes interactivos funcionan en la aplicación se elaboró el siguiente modelo conceptual:

⁸Término utilizado para describir sitios corporativos

⁹Término utilizado para productos como aplicaciones para móviles

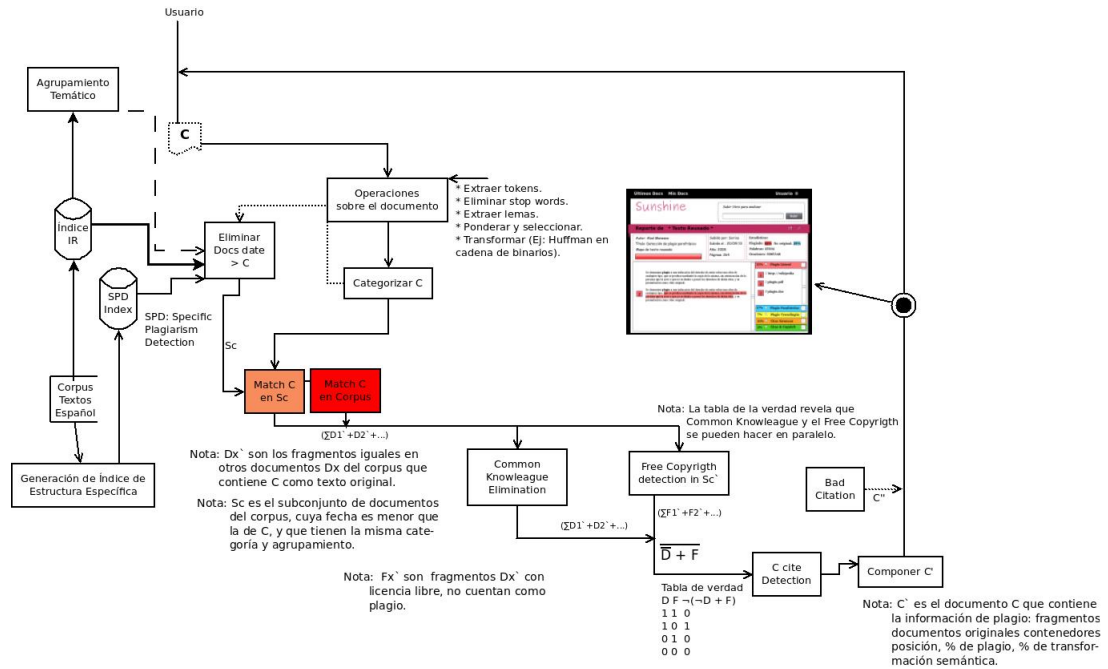


Figura 9: Diagrama de subprocesos del módulo de Sunshine de Texto Reusado

Conocer perfectamente el modelo conceptual de un sitio permite hacer un consistente diseño de decisiones. No importa donde el elemento de contenido u objeto esté ubicado, lo que importa es que el sitio se comporte correctamente.[5]

Dado que la visualización de la revisión de plagio contendría gran cantidad de información representa un gran reto crear una experiencia de usuario basada en el entendimiento de los objetivos del sitio y las necesidades del usuario, para ello debe utilizarse una nomenclatura concerniente al tema que sea entendible de manera que los usuarios sean capaces de moverse por el sitio encontrando su propio camino a través de la arquitectura. Por esta razón es necesario utilizar un lenguaje sugerente. Jesse James Garret usa un término llamado **vocabulario controlado** que persigue asegurar que todo el mundo esté hablando en el lenguaje del usuario. A continuación se relacionan los términos empleados como etiquetas en el negocio:

- Mapa de texto reusado: Esta especie de barra de progresión permite ver el % de texto reusado que posee el documento, mostrando los distintivos colores de cada tipo de plagio.
- Estadísticas: mostrará las estadísticas del documento provenientes del análisis del texto reusado.
- Plagio: % del documento plagiado de cualquier tipo posible.

- Otros: % del documento que no es plagio pero tampoco es original. (Ej: Bad Citation o Copyleft)
- Original: % de texto original redactado por el autor o autores.
- Palabras: Cantidad de palabras del documento original.
- Oraciones: Cantidad total de oraciones del documento original.

Además en la vista del usuario **avanzado** (*o investigador*) se utilizan las etiquetas:

- N-gramas: Cantidad total de N-gramas del documento de acuerdo al número de grams bajo el cual se realizó el análisis.
- Autoplagio: Permite visualizar en la vista y en el árbol solo los fragmentos procedentes de autoplagio (self – plagiarism) ya sean plagio literal, parafrástico o multilingüe.

Diseño de la Información/ Interfaz y Navegación (Cuarto plano o Skeleton Plane):

Esta etapa define la ubicación de botones, controles, fotos, texto u otros elementos. Debe ser diseñada para optimizar dicha alineación para lograr un mejor efecto y eficiencia, de manera que el usuario pueda encontrar lo que necesita dónde lo necesita.[5]

Aquí se destacan dos aspectos fundamentales de la UX. Por un lado el **diseño de la interfaz** proveerá al usuario de la habilidad para hacer cosas, por otro, el **diseño de la navegación** de la posibilidad de ir a distintos lugares. Ambos no pueden funcionar de manera exitosa sin un buen diseño de la información.[5]

Teniendo en cuenta el diseño de interfaz, para Sunshine se desean escoger los componentes visuales adecuados, para ello se evaluarán las últimas tendencias estudiadas en el estudio de homólogos; de manera que el usuario sepa encontrar lo que es realmente importante en un sitio de esta complejidad. A continuación se relacionan los componentes seleccionados destinados a lograr una mejor usabilidad de la interfaz:

- Etiqueta (*label*): Utilizadas para nombrar toda la información del negocio.
- Caja de chequeo (*checkbox*): Permite el marcado de múltiples opciones.
- Popup menú (*popup menu*): Mostrará las opciones del perfil del usuario autenticado.
- Menú desplegable: Actúa como acordeón, mostrando las fuentes detectadas en cada tipo de plagio.
- *Browse* (unión de un campo de texto y un botón): Este complemento permite accionar la vista de selección de archivos.

La navegación de cualquier sitio debe cumplir simultáneamente las siguientes metas:

- Primero, debe proveerle al usuario los medios para ir de un punto al otro del sitio. Los elementos de navegación deben ser seleccionados para facilitar comportamiento real del usuario.
- Segundo, el diseño de navegación debe comunicar la relación entre los elementos que contiene. No es suficiente facilitar una lista de links. ¿Qué tienen que ver unos con otros? ¿Cual es la diferencia de relevancia entre ellos?.
- Tercero, el diseño de navegación debe contener la comunicación entre sus contenidos y la página que actualmente esta viendo el usuario. ¿Qué tienen que ver estos con lo que esta buscando?[5]

En los inicios de la web la herramienta utilizada para diagramar la estructura del sitio era conocida como mapa del sitio, pero dado que así también se le conoce a un tipo de herramienta de navegación dentro del sitio, actualmente se prefiere el término **diagrama de arquitectura**Garrett [5]. A continuación se muestra dicho diagrama para el caso de Sunshine:

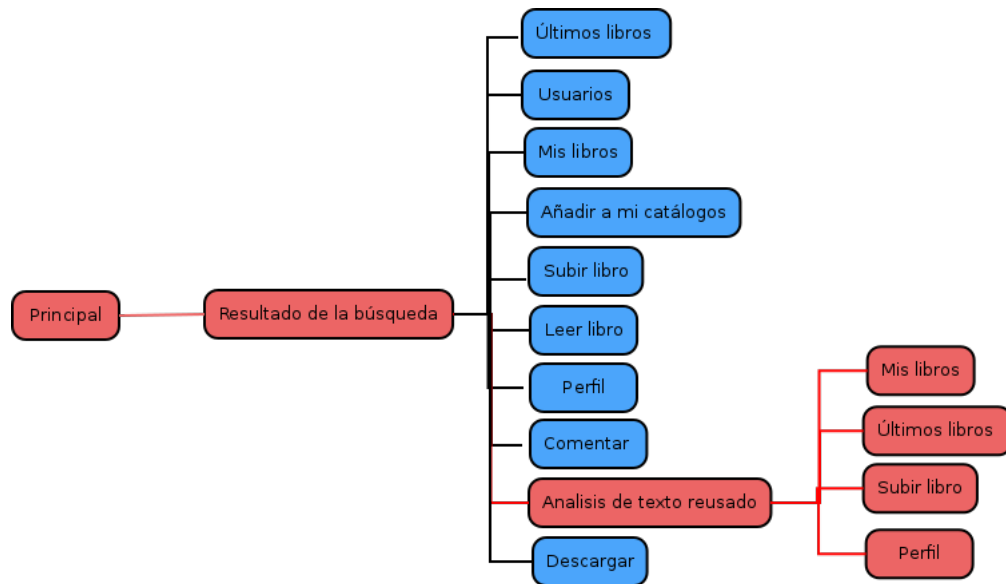


Figura 10: Diagrama de arquitectura mostrando el camino hacia las funcionalidades de plagio

El diseño de la información involucra la agrupación de las distintas piezas del rompecabezas concernientes a la información propia del negocio. La clave radica en ubicar los elementos de la información en categorías que reflejen cómo piensa el usuario. Por ejemplo en el caso de Sunshine se puede organizar la información de la siguiente manera:

- Tipos de plagio:
 - Plagio Literal (Verbatim Plag)
 - Plagio Parafrástico (Paraphrase Plag)
 - Plagio Translingüe (Crosslang Plag)
 - Citación Errónea (Bad Citation)
 - Citación y Copyleft (Citation & Copyleft)
- Datos del perfil de usuario:
 - Nombre
 - Usuario
 - Email(correo electrónico)
 - Profesión
 - Direccion particular
 - Teléfono
 - Centro de trabajo

El diseño de los mensajes de error y de la ayuda del sitio, constituyen problemas del diseño de la información; el objetivo fundamental de Sunshine a vencer es que realmente los usuarios lean estas instrucciones.[5]

El reporte de plagio ha sido diseñado de manera que los usuarios puedan ubicar fácilmente lo que están buscando o necesitan en un momento determinado, esta técnica, conocida como **wayfinding** [5], se representa a través del marcado de los diferentes tipos de plagio en colores diferentes por lo que resulta mas fácil la familiarización con estos elementos y luego de un primer contacto con ellos la búsqueda en el documento resulta más especilizada.

- Rojo: Señala el tipo de plagio literal.
- Azul(cyan): Señala el tipo de plagio parafrástico.
- Amarillo: Señala el tipo de plagio translingüe.
- Naranja: Señala las citas mal referenciadas.
- Verde: Señala las citas correctas y fragmentos copyleft reutilizados con o sin variaciones de forma correcta (contemplando el caso de las obras derivadas que deben ser citadas al inicio, pero se consideran piezas originales de texto).

Se propone para reporte de plagio un estándar centrado en una visualización de los elementos de plagio detectados a través del sistema. A continuación se muestra el esquema de la página o **wireframe** utilizado para el diseño:

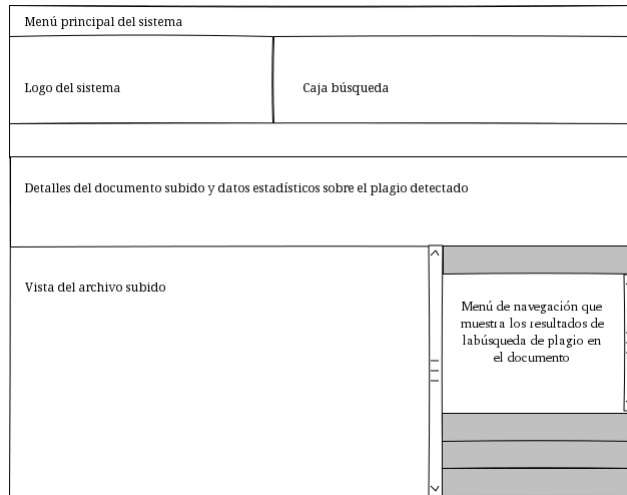


Figura 11: Wireframe

Diseño Visual(Quinto plano o Surface Plane):

En esta capa se trata con el diseño sensorial y la presentación lógica de los elementos ubicados que componen el *nivel de esquema*. Por ejemplo, a través del cuidado del diseño de la información, se determina como deben agruparse y organizar los elementos de información en la página y por ende el diseño visual determina como estos elementos deben ser presentados visualmente. [5]

De acuerdo a lo observado en el estudio de homólogos se desea crear un estándar visual para sistemas de plagio, para ello deben tenerse en cuenta los siguientes aspectos:

- El reporte a mostrar debe integrar:
 - % de cada tipo de texto reusado (plagio o no) encontrado en el documento.
 - Vista del documento con el texto señalado que procede de las fuentes sospechosas, esta señalización debe ser distintiva para cada tipo de plagio.
 - Estadísticas referentes al % de cada tipo de plagio, el % que no es plagio pero tampoco es original, la cantidad de palabras del documento original y cantidad de oraciones del documento original.
- Debe tenerse en cuenta que este tipo de sistemas debe tener una exquisita organización de la información puesto que la misma es abundante en estos reportes y se puede correr el riesgo que el usuario se sienta abrumado y perdido.
- Como se emplearán señalizaciones en el contenido del documento para destacar la atención de usuario, el resto de la vista debe pensarse con colores que creen un ambiente simple para no caer en extravagancias innecesarias.

El color puede ser una de las vías mas efectivas para comunicar la identidad de una marca. Algunas marcas están tan estrechamente ligadas a un color que es difícil pensar en la compañía sin que venga a la mente el color que la distingue, ejemplo de ello son Coca Cola, Mc Donald, eBay, entre otras. En este caso, teniendo en cuenta los colores y la tipografía, se tomaron en cuenta los siguientes elementos:

- Colores más opacos son utilizados como fondo para elementos que se desean destacar (blanco y gris).
- Contraste formado por una especie de marco oscuro en el que se resalta el centro de la página o el área de atención.
- Para el nombre del sistema se utilizó la letra Purisa pues provoca igualmente sensación académica o de modernidad .

A continuación se muestra una plantilla donde se observan los contrastes de color de la página:

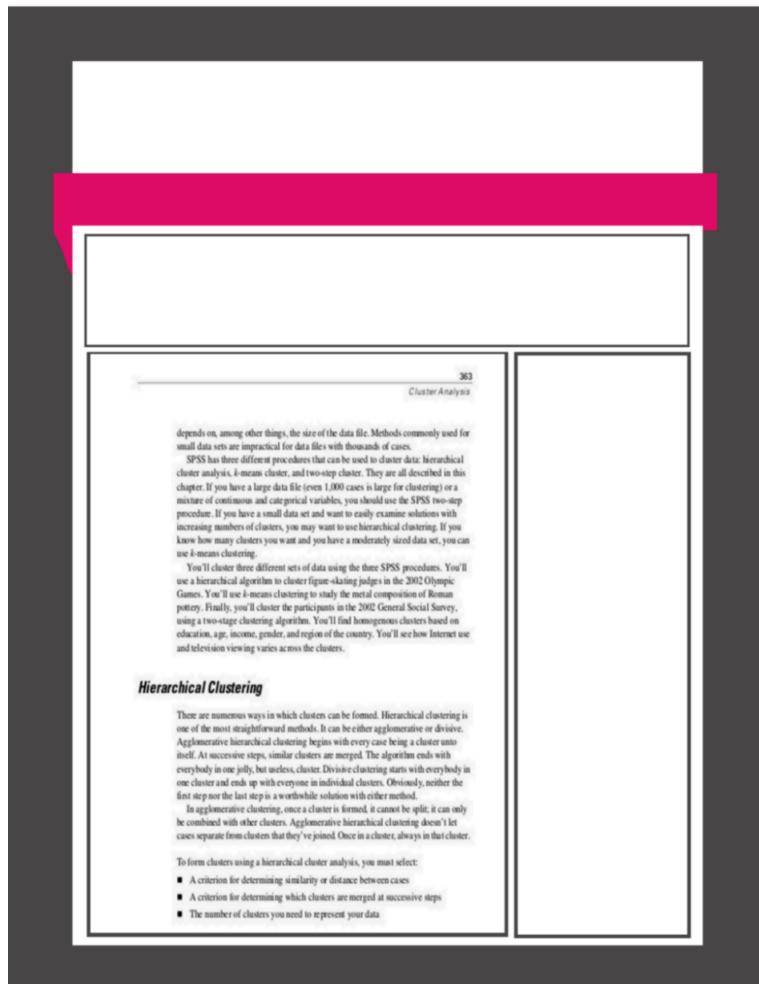


Figura 12: Contraste de color para la vista de Reporte de plagio


Resultados

Se diseñó la vista de plagio para Sunshine reuniendo los aspectos descritos anteriormente, tal como se observa:

1

Textos Recientes Mis textos

2

Usuario 

3

Sunshine

Subir texto para analizar:

Subir

4

Reporte de * Texto Reusado *

5

Estadísticas:

Plagiado: 42% No original: 29%

Palabras: 25634


Oraciones: 6985348

6

Autor: Abel Meneses

Título: Detección de plagio parafrásico

Mapa de texto reusado:



Subido por: Sorice

Subido el : 20/09/13

Año: 2008

Páginas: 264

7

Se denomina **plagio** a una infracción del derecho de autor sobre una obra de cualquier tipo, que se produce mediante la copia de la misma, sin autorización de la persona que la creó o que es su dueña o posee los derechos de dicha obra, y su presentación como obra original.

3

Se denomina **plagio** a una infracción del derecho de autor sobre una obra de cualquier tipo, que se produce mediante la copia de la misma, sin autorización de la persona que la creó o que es su dueña o posee los derechos de dicha obra, y su presentación como obra original.

8

15% ☐ Plagio Literal

1 ☐ http://wikipedia

2 ☐ plagio.pdf

3 ☒ plagio.doc

17% ☐ Plagio Parafrásico

7% ☐ Plagio Translingüe

10% ☐ Citas Erróneas

2% ☐ Citas & Copyleft

Figura 13: Prototipo para la detección de plagio en Sunshine(Vista para usuarios novatos)

22

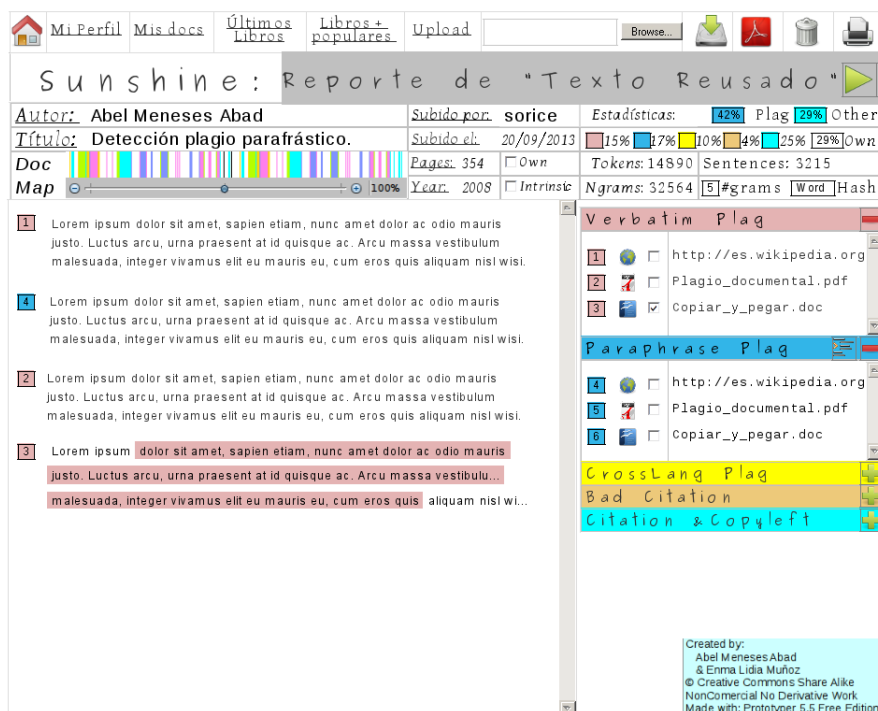


Figura 14: Prototipo para la detección de plagio en Sunshine(Vista para usuarios expertos)

Descripción de los elementos visuales de las vistas:

1. Opciones del menú de navegación principal de la aplicación.
2. Nombre del usuario autenticado, así como la imagen del perfil del mismo.
3. Selección del(de los) archivo(s) a cargar para analizar.
4. Menú con las opciones a ejecutar referentes al análisis de plagio de texto cargado: guardar reporte de plagio, ayuda, cambiar de la vista básica a la avanzada y viceversa.
5. Datos sobre el documento subido y algunos elementos sobre el análisis de plagio.
6. La vista avanzada muestra además más datos relativos al análisis de plagio.
7. En esta área se muestra la vista del documento, en el mismo se señalará el texto encontrado en las diferentes fuentes indicado en el color definido para cada tipo de plagio.
8. Menú lateral que funciona como una especie de acordeón y agrupa las fuentes distribuidas por tipos de plagio.

Conclusiones

Existen pocos análisis científicos orientados al análisis de las aplicaciones para la detección de plagio.[17],[27],[26] En cambio es posible encontrar artículos dispersos valorando aplicaciones de forma aislada como: Antiplag[18],

CitePlag[7], firefox plugin[4], iPlag[2], PlaDeS[25], Turnitin[16]. Y finalmente otros pocos artículos hacen un análisis breve de varios de ellos como parte no esencial de su estudio.[20] En particular el número de sistemas desarrollados, los innumerables métodos para la detección de texto reusado y la audiencia en aumento de este tipo de aplicación requieren un análisis y estandarización de la experiencia de usuario de los mismos, algo que explícitamente no ha sido tratado en la literatura internacional. Teniendo en cuenta lo anteriormente planteado y tomando como referencia los sistemas estudiados se obtuvo como resultado lo siguiente:

- En todos los sistemas encontrados aunque se marca en el documento el texto sospechoso de plagio con color, no se indica el tipo de plagio determinado para cada caso.
- SeeSources.com y Plagiarism-detect.com son sistemas web que realizan revisión de plagio a partir de un único documento subido o de una porción de texto copiado, comparándolo con las fuentes en internet. Los resultados que muestran las ocurrencias encontradas de cada fuente, por lo que no resulta visualmente atractivo.
- Viper aunque posee las características de los sistemas anteriores en la vista del análisis de plagio muestra comparación en paralelo entre el texto subido y la fuente, marcando las partes sospechosas (*side-by-side comparison*). También realizan dicha comparación Turnitin, JPlag, SafeAssign, CrossCheck y CitePlag; de estos resultan una buena opción a tener en cuenta, las propuestas visuales de CrossCheck y Turnitin.
- De la vista del módulo OriginalityCheck de Turnitin, resulta visualmente interesante el menú lateral derecho nombrado *Match Overview*, el cual describe el porcentaje de ocurrencias encontradas de cada fuente en el texto revisado. Es por ello que para el sistema de detección de plagio de Sunshine se desea crear una versión de este componente adaptado a los distintos tipos de plagio.
- Se toma para Sunshine la idea de SafeAssign y CrossCheck de encabezar la vista del reporte de plagio con los detalles del documento subido, incluyendo además los datos estadísticos sobre el análisis del plagio para la vista avanzada.
- Teniendo en cuenta Turnitin, Safeassign, Viper y CrossCheck se considera más práctico para el usuario usar solo una vista que centralice tanto el documento subido con el texto sospechoso marcado, así como el resto de los detalles de los tipos de plagio. Este principio se utiliza en el diseño empleado en Sunshine.
- Puesto que Sunshine es un repositorio institucional, utilizará también para las búsquedas su propia base de datos de documentos almacenados tal como lo hacen los sistemas Turnitin, JPlag, Coremo, CitePlag; sistemas enfocados al entorno educativo.

Se logró establecer en la revisión los elementos comunes, estos han sido aceptados por los desarrolladores, y no tienen complejidad computacional. Son muy sencillos permitiendo enlazar los servicios y complacer a los usuarios.

Referencias Bibliográficas

- [1] AllAnswersLtd (2014). Example scan.
- [2] Alzahrani, S., Salim, N., Abraham, A., and Palade, V. (2011). iPlag: Intelligent Plagiarism Reasoner in Scientific Publications. In *World Congress on Information and Communication Technologies*, pages 1–6.
- [3] Blackboard.Inc (2013). Safeassign - blackboard help.
- [4] Chiu, S., Uysal, I., and Croft, W. B. (2010). Evaluating Text Reuse Discovery on the Web. pages 299–303.
- [5] Garrett, J. J. (2011). *THE ELEMENTS OF USER EXPERIENCE*. Second edition edition.
- [6] Gipp, B. (2013). *Citation-based Plagiarism Detection*. Phd thesis, Otto-von-Guericke-University Magdeburg, Magdeburg.
- [7] Gipp, B. and Meuschke, N. (2011). Citation Pattern Matching Algorithms for Citation-based Plagiarism Detection: Greedy Citation Tiling, Citation Chunking and Longest Common Citation Sequence. In *DocEng’11*, pages 249–258, Mountain View, California, USA. ACM Press.
- [8] Gipp, B., Meuschke, N., Breitingner, C., Lipinski, M., and Nümberger, A. (2013). Demonstration of Citation Pattern Analysis for Plagiarism Detection. In *SIGIR’13*, pages 1119–1120, Dublin, Ireland. ACM Press.
- [9] Grman, J. and Ravas, R. (2011a). Improved implementation for finding text similarities in large collections of data. *PAN’2011*, page 6.
- [10] Grman, J. and Ravas, R. (2011b). Improved implementation for finding text similarities in large collections of data. In *PAN’2011*, page 18.
- [11] IParadigms, L. and Turnitin (2012a). Turnitin - GradeMark.
- [12] IParadigms, L. and Turnitin (2012b). Turnitin - Inicio.
- [13] IParadigms, L. and Turnitin (2012c). Turnitin - OriginalityCheck.
- [14] IParadigms, L. and Turnitin (2012d). Turnitin - PeerMark.
- [15] iThenticate (2009). CrossCheck user manual.
- [16] Jones, K. O. and Moore, T. A. (2010). Turnitin is not the primary weapon in the campaign against plagiarism. In *CompSysTech ’10, 11th International Conference on Computer Systems and Technologies*, page 425, New York, New York, USA. ACM Press.
- [17] Kakkonen, T. and Mozgovoy, M. (2010). Hermetic and web plagiarism detection systems for student essays— an evaluation of the state-of-the-art. 42(2):135–159.

- [18] Kakkonen, T. and Myller, N. (2009). AntiPlag - A sampling-based tool for plagiarism detection in student texts. page 8.
- [19] Karjoo, M. (2009). How Does SafeAssign Work - SafeAssign Wiki - Confluence.
- [20] Kashkur, M., Parshutin, S., and Borisov, A. (2010). Research into Plagiarism Cases and Plagiarism Detection Methods. *Scientific Journal of Riga Technical University*, 44:139–145.
- [21] KIT (2013a). Jplag - detection software plagiarism.
- [22] KIT (2013b). Search results.
- [23] Knight, E. (2007). Safeassign sample report.
- [24] Kravjar, J. (2012). Barrier to thriving plagiarism. In *5th International Plagiarism Conference*, page 11. Slovak Centre of Scientific and Technical information.
- [25] Kucecka, T. (2011). Obfuscating Plagiarism Detection - Vulnerabilities and Solutions. In *CompSysTech'11 International Conference on Computer Systems and Technologies*, pages 423–428, Viena, Austria. ACM Press.
- [26] Lancaster, T. and Culwin, F. (2007). Classifications of plagiarism detection engines. page 16.
- [27] Maurer, H., Kappe, F., and Zaka, B. (2006). Plagiarism - A Survey. *Journal of Universal Computer Science*, 12(8):1050–1084.
- [28] Mediaphor, S. E. A. (2013). Plagiarism-finder.
- [29] Miranda, C. (2013). *A Study on Plagiarism Detection and Plagiarism Direction Identification Using Natural Language Processing Techniques*. Ph.d. thesis, University of Wolverhampton, Wolverhampton.
- [30] Nelson, M. (2006). Academic integrity. Imagen.
- [31] Potthast, M., Hagen, M., Gollub, T., Tippmann, M., Kiesel, J., Rosso, P., Stammatatos, E., and Stein, B. (2013). Overview of the 5th International Competition on Plagiarism Detection. In Forner, P., Navigli, R., and Tufis, D., editors, *PAN'2013*, pages 23–26.
- [32] Rodriguez-Rorrejón, D. (2013). Sobre CoReMo system.
- [33] Rodríguez Torrejón, D. A. (2013). CoReMo 2.1 Plagiarism Detector. In *PAN'2013*, page 1.
- [34] SCIPlore Knowledge Discovery (2013). CitePlag example report.
- [35] Scott, B. (2013). Free online plagiarism checkers and duplicate content detectors.

- [36] Segaran, T. (2007). *Programming Collective Intelligence*. O'Reilly, first edit edition.
- [37] Viper (2012). Plagiarism free software?

Anexos

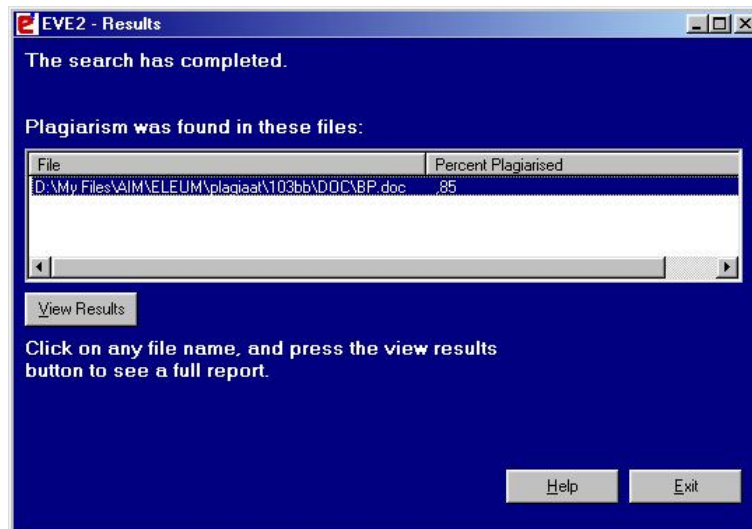


Figura 15: Eve2 (1)

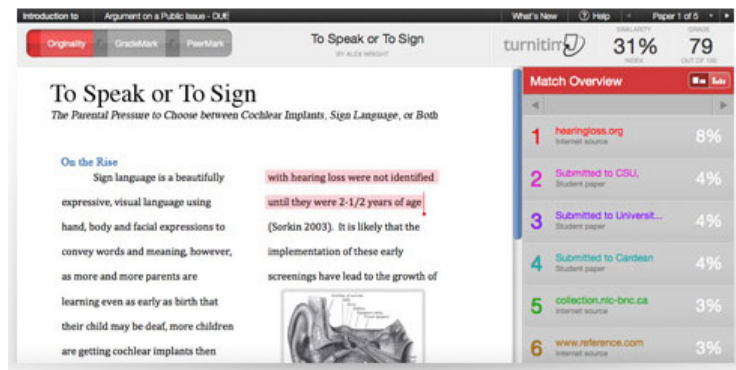


Figura 16: OriginalityCheck

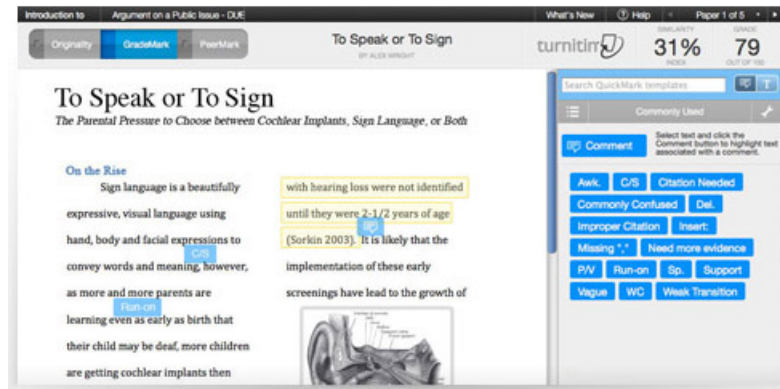


Figura 17: Grademark

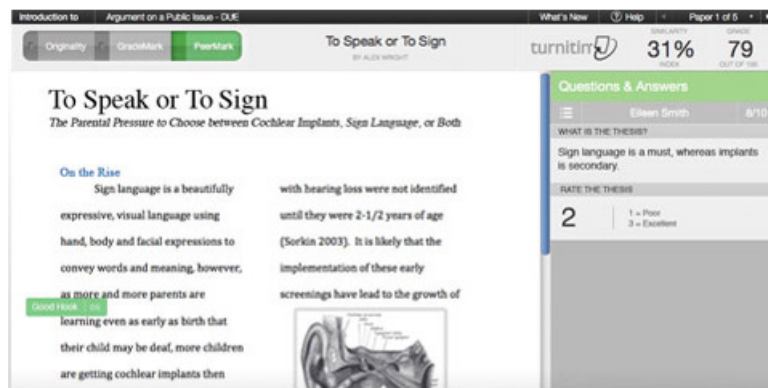


Figura 18: PeerMark

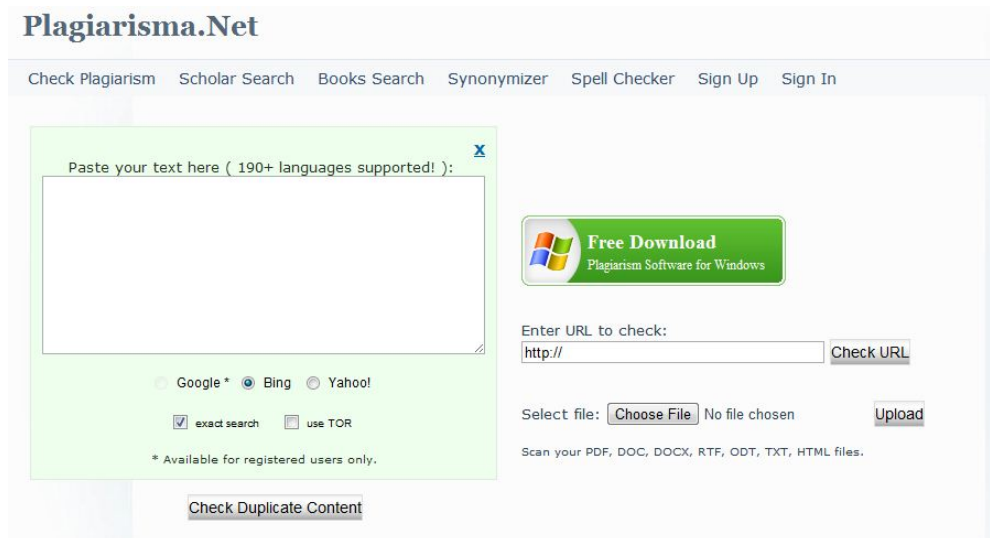


Figura 19: Página oficial de Plagiarisma.net

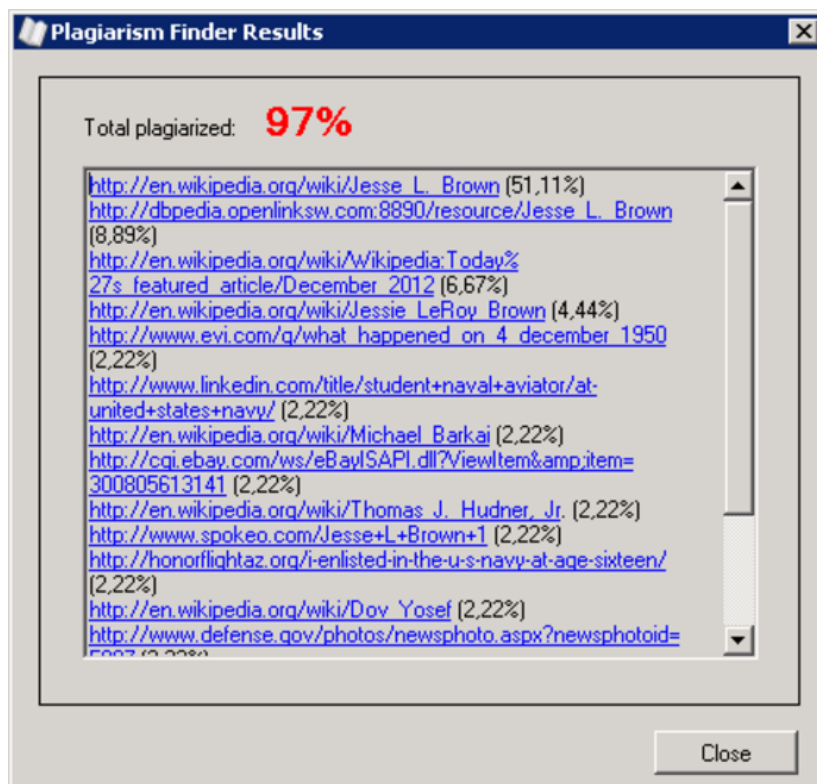


Figura 20: Reporte de resultados de Plagiarism Finder