

# Editor d'ajuda a la correcció de redaccions escrites en anglès

Sergi Oliva Giménez

**Resum**— Aquest projecte va consistir en el desenvolupament d'una aplicació web que permetrà als professors d'anglès poder connectar-se a un servidor, i pujar els treballs que els alumnes els hi enviïn. Mitjançant una sèrie d'etiquetes podran corregir i afegir comentaris a les redaccions escrites pels alumnes. Així mateix, guardar-les en una base de dades per a una posterior anàlisi i enviament, una vegada corregides, mitjançant un correu electrònic als alumnes.

**Paraules clau**— Corrector redaccions, TinyMCE, Professors d'anglès, Etiquetes correctores.

**Abstract**— This project consisted in the development of a web application that will allow English teachers to connect to a server, upload the writings of the students and provide a set of tools to correct and add comments to the students' writings. Allowing the teachers to save them in a database for a later analysis and send them corrected by email back to the students.

**Index Terms**— Writing editor, TinyMCE, English teachers, Corrective labels.



## 1 INTRODUCCIÓ

AQUEST projecte neix de la problemàtica dels professors del departament de Filologia Anglesa i de Germanística, a l'hora de corregir les diferents redaccions dels alumnes. Actualment, els alumnes els hi entreguen les redaccions en mà.

Posteriorment, els professors les corregeixen i les lliuren als alumnes quan es tornen a veure a classe. D'aquesta manera, és complicat mantenir un control sobre els tipus d'errors més comuns realitzats pels alumnes, i la manera de corregir dels professors. El que vol proporcionar el projecte és una eina per aconseguir una major automatització i estandardització a l'hora de corregir. També, mantenir un millor control sobre els tipus d'errors que realitzen els alumnes i emmagatzemar les redaccions en una base de dades.

A continuació detallarem:

- Una breu descripció del treball realitzat.
- Els objectius proposats a l'hora de desenvolupar l'eina.
- L'estat de l'art per avaluar el que es troba actualment en el mercat, i així tenir unes bones referències.
- Els requeriments inicials per a desenvolupar l'eina, els requeriments que han sorgit posteriorment i els diagrames de casos d'ús.

- Les eines utilitzades en la realització del projecte.
- La metodologia que s'ha triat a l'hora de treballar en aquest projecte.
- La planificació que s'ha emprat a l'hora de realitzar el projecte.
- El disseny del projecte indicant els diferents inconvenients trobats.
- El desenvolupament del projecte amb els diferents inconvenients trobats i els tests realitzats.
- Exposició de resultats.
- Les conclusions.
- Línies futures a seguir i a millorar.
- Agraïments.
- La bibliografia.
- Una secció d'apèndix.

## 2 DESCRIPCIÓ DEL TREBALL

S'ha desenvolupat una eina (editor) que permet als diferents professors d'anglès la correcció de treballs escrits. L'eina treballa amb un fitxer d'entrada amb format html que vindrà convertit d'un Word[4] o LibreOffice[5], ja que els alumnes entreguen les seves redaccions normalment amb aquests formats. Es retornarà la versió corregida a l'alumne en el format html, però amb les anotacions pertinents. Tanmateix, tant el text corregit com el text original es guardaran en una base de dades, per a la posterior creació d'un corpus d'estudiants. L'eina està basada en un editor de text, format per un conjunt d'etiquetes agrupades per categories: gramàtica, vocabulari, estil, etc. Cada categoria té un llistat de subcategories. Cada text està

- E-mail de contacte: [sergi.oliva@e-campus.uab.cat](mailto:sergi.oliva@e-campus.uab.cat)
- Menció realitzada: *Enginyeria del Software*.
- Treball tutoritzat per: *Marta Prim Sabria (Microelectrònica i Sist. el i Sistemes Electrònics)*
- Curs 2018/19

escrit per un alumne i corregit per una professora. Aquesta informació es mantindrà amb un codi perquè la font ha de ser anònima.

### 3 OBJECTIUS

En l'apartat d'objectius tenim dues parts diferenciades:

- Pels alumnes, quins avantatges els hi pots suposar que els professors utilitzin l'eina.
- Pels professors, què els hi suposarà l'ús de l'eina tant en l'àmbit immediat com a un nivell posterior al seu ús.

#### 3.1 Alumnes

De cara als alumnes els objectius que podem destacar són els següents:

- Hi haurà una major estandardització a l'hora de rebre treballs corregits. Considerant que si en aquests hi ha errors, les correccions dels professors estaran formades per un conjunt d'etiquetes on cadascuna té una definició del tipus d'error. Aquestes etiquetes seran comunes entre professors. Per tant, si l'alumne canvia de professor, sabrà quin és exactament l'error que ha realitzat.
- Tot es realitzarà amb tipografia d'ordinador on no hi haurà dubtes d'allò que hi ha escrit, perquè de vegades, amb la tipografia a mà, generava dubtes entre els alumnes d'allò que havia escrit.
- També, hi haurà una major velocitat a l'hora de rebre les puntuacions i correccions dels treballs. Abans al fer-ho en paper, era necessari que l'alumne i la professora estiguessin presents, tant per rebre els documents com per lliurar-los.

#### 3.2 Professors

Respecte als professors, els factors més importants que podem destacar són:

- Una major velocitat a l'hora de corregir. Per tal d'indicar un error, només caldrà marcar el text i prémer un botó. L'eina agafarà el text i el marcarà amb tags indicant el tipus d'error, sense que el professor l'hagi d'escriure.
- Més simplicitat, els errors estaran agrupats per un conjunt d'etiquetes on serà més fàcil detectar el tipus d'error.
- Una millor organització entre el grup de professorat, atès que aquestes etiquetes seran comunes entre ells.
- Un corrector que indicarà quan una paraula està mal escrita, cosa que afavorirà que els professors es puguin focalitzar en detectar altres tipus d'errors.
- Posteriorment, aquestes redaccions es podran guardar en una base de dades, tant les originals com les corregides, enviant-les a l'alumne que les ha lliurat. L'emmagatzematge d'aquests treballs, suposarà una bona eina, ja que més endavant es podran analitzar, veient quines són les errades més comunes. D'aquesta manera, els professors podran enfocar els propers exercicis per tal de

corregir aquests tipus d'errors.

## 4 ESTAT DE L'ART

Actualment, en el mercat hi ha diversos tipus de correctors, tant aplicacions web com aplicacions d'escriptori. Per tal d'avaluar la necessitat d'implementar aquesta eina, analitzarem els diferents punts positius i negatius de les alternatives de mercat i analitzarem la proposta:

### 4.1 Eines web

En aquest apartat hi podem destacar les següents eines: Grammarly[1], Reverso[2], OnlineCorrection[3].

Pel que fa a Grammarly és una eina web que té els següents avantatges:

- Permet corregir errors de gramàtica, ortogràfics, de puntuació i té un corrector ortogràfic contextual en la seva versió gratuïta.
- També pot ser utilitzada com a extensió del navegador.
- Té una versió de pagament més completa que inclou algunes funcionalitats, com ara, el corrector ortogràfic de frases.

Com a desavantatges importants podem destacar:

- No pot carregar els documents directament.
- No pot enviar els documents de tornada als alumnes.
- No permet modificar els errors indicant el tipus que volem.

Seguint amb les diferents eines tenim Reverso. En quant a avantatges:

- També té versió d'extensió de navegador a part de corrector online.
- Permet corregir errors gramaticals, proporcionant una varietat d'opcions per a escollir el tipus d'error.

En relació als desavantatges:

- No pot carregar els documents directament.
- No pot enviar els documents de tornada als alumnes.
- No permet modificar els errors indicant el tipus que volem.

Per acabar, trobem OnlineCorrection.com. D'avantatges trobem:

- El seu format de correcció és el que més s'assimila amb el que es vol realitzar.
- Marca les paraules o conjunt de paraules amb diferents colors, indicant si és un error d'escriptura, gramatical, etc. I ens indica els diferents suggeriments per a corregir cadascun d'ells.

Com a desavantatges:

- No té extensió de navegador.
- No pot carregar els documents directament.
- No pot enviar els documents de tornada als

alumnes.

- No permet modificar els errors indicant el tipus que volem.

En resum, són eines on marquen errors amb diferents colors indicant el tipus de cadascun d'ells. Comparat amb la nostra eina, caps d'ells proporciona l'opció de carregar directament un document, de guardar-lo i enviar-lo a l'alumne, ni d'afegir un conjunt d'etiquetes modificades.

## 4.2 Eines d'escriptori

D'eines d'escriptori, podem destacar les bàsiques com ara els editors de texts en els quals els alumnes realitzaran les redaccions: Microsoft Word, LibreOffice Writer, amb les seves extensions pertinents, com ara Ginger Grammar.

Microsoft Word és l'eina d'edició de documents de Microsoft, és una de les més utilitzades, i incorpora un corrector bàsic, tot i que hi ha extensions que li proporcionen millores a l'hora de la correcció.

També trobem el LibreOffice Writer que és l'editor de documents en versió OpenSource. Com l'eina anterior, incorpora un corrector molt bàsic, i també se li poden afegir extensions per fer-lo més complert.

Comparat amb la nostra eina, els hi faltaria generar les etiquetes i marcar els errors de diferents colors. Poder guardar els documents en una base de dades i enviar-los als alumnes. Aconseguir que fos accessible des de qualsevol lloc, ja que la nostra eina a l'estar allotjada al servidor, permet que els professors es connectin des de qualsevol ordinador.

## 5 REQUERIMENTS

Aquest apartat estarà dividit entre els requeriments inicials que vam obtenir i els requeriments que s'han anat obtenint al llarg del disseny i desenvolupament de l'aplicació:

### 5.1 Requeriments inicials

Aquí trobarem els primers requeriments, abans de començar la realització de l'eina:

- [R1] Els diferents professors han de poder fer login al sistema amb un usuari i contrasenya.
- [R2] Els documents a importar han de ser en format Word o OpenOffice.
- [R3] L'eina ha d'incorporar un corrector simple que identifiqui errors bàsics.
- [R4] L'eina ha de permetre l'ús de comentaris tipus text als professors.
- [R5] L'eina ha de permetre guardar els documents amb el mateix format que els importa.
- [R6] L'eina ha de permetre enviar els documents una vegada corregits als alumnes corresponents.
- [R7] L'eina ha de permetre l'enviament del fitxer

original a una base de dades.

- [R8] L'eina ha de permetre l'enviament del fitxer corregit a una base de dades.
- [R9] L'eina ha de permetre la generació d'etiquetes de manera simple.

### 5.2 Requeriments posteriors

En aquest apartat indicarem els requisits que s'han desenvolupat posteriorment, també si s'han modificat els originals:

- [R2] No necessari.
- [R5] L'eina ha de permetre guardar els documents en un format llegible per un usuari bàsic.
- [R10] Les etiquetes hauran d'estar agrupades en subcategories.
- [R11] Els fitxers s'hauran d'agrupar segons l'any al sistema gestor de la base de dades.

En el cas del [R2], es va indicar posteriorment, que no seria necessari que els documents estiguessin en aquest format, ja que els professors els podrien convertir en un format que a l'eina li resultes llegible.

En el cas del [R5], el que interessa és que els usuaris puguin llegir el document que han enviat amb les correccions pertinents per a saber on s'han equivocat i quin tipus d'error tenien.

En el cas del [R10] s'ha incorporat perquè la categoria de gramàtica incorporava 19 tipus d'etiquetes i el desplegable quedava massa llarg.

En el cas del [R11] s'ha decidit de moment incloure una separació per any en relació als documents i als alumnes, i així tenir una classificació.

### 5.3 Diagrama de casos d'ús

S'ha realitzat un diagrama de casos d'ús per l'estudiant, pel professor i per l'administrador, on es pot veure les diferents funcions que realitzaran cadascun d'ells.

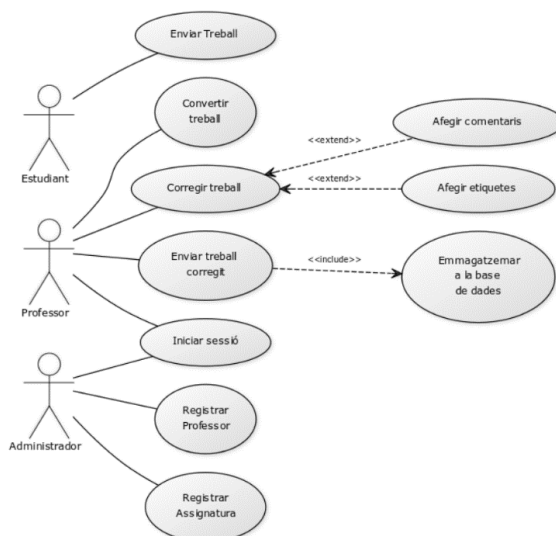


Fig. 1. Diagrama de casos d'ús.

## 6 EINES UTILITZADES

Definirem les diferents eines utilitzades per apartats:

### 6.1 Servidor

Com a servidor s'ha utilitzat un servidor XAMPP. Del qual hem agafat el següent:

- Servidor web Apache. És un servidor estàndard del qual hi ha molta informació i és el més utilitzat, proporcionava l'entorn necessari per a la nostra eina.
- MySQL. De servidor de base de dades hem utilitzat el MySQL, ja que ens venia integrat amb el XAMPP. El que necessitàvem era un gestor de base de dades estàndard per emmagatzemar texts, números, etc.
- Mercury. De servidor de correu hem utilitzat Mercury, perquè és l'opció que ens proporcionava XAMPP i funciona correctament amb la llibreria de PHP, utilitzada per a poder enviar correu als diferents destinataris.

### 6.2 Servidor SMTP

A l'hora de realitzar enviaments a diferents alumnes, s'ha apostat per utilitzar un servidor SMTP[20], perquè així estaria tot centralitzat. Escollir un servidor SMTP era necessari, atès que la llibreria php la necessitava per a funcionar correctament. El servidor SMTP escollit, ha estat un de gratuït, anomenat Mailjet[18].

### 6.3 Servidor de versions

Per a mantenir un control del projecte, s'ha anat treballant amb un repositori extern, en el qual s'hi anaven pujant tots els canvis. Tant per si es treballava des d'un ordinador diferent poder seguir treballant o simplement per a no perdre el progrés realitzat. Aquest servidor ha estat el bitbucket[23].

### 6.4 Altres eines

Per a realitzar la transformació massiva de documents doc, docx, odt, etc. a format html compatible amb el nostre editor, s'ha utilitzat "MultiDoc Converter". Aquest permet seleccionar una carpeta i convertir tots els fitxers correctament a html. S'ha escollit aquesta eina en front d'altres perquè és una eina gratuïta, permet transformar massivament els documents d'una carpeta a un format html que manté el format enviat pels alumnes. També és una eina intuïtiva i senzilla, cosa que els professors agraeixen.

## 7 METODOLOGIA

La metodologia considerada en aquest treball ha estat la de Desenvolupament iteratiu i incremental[8]. És una metodologia basada en diferents blocs temporals anomenats iteracions. Cada iteració es pot entendre com un miniprojecte que conforma un producte final. Així doncs, el client podrà obtenir una millor visió de com va evolucionant el projecte i si aquest necessita algun canvi o alguna millora que no estava contemplada. Es va triar aquesta metodologia perquè es volia partir d'un projecte

base i anar interant posteriorment amb miniprojectes per a complir els objectius proposats a l'inici.

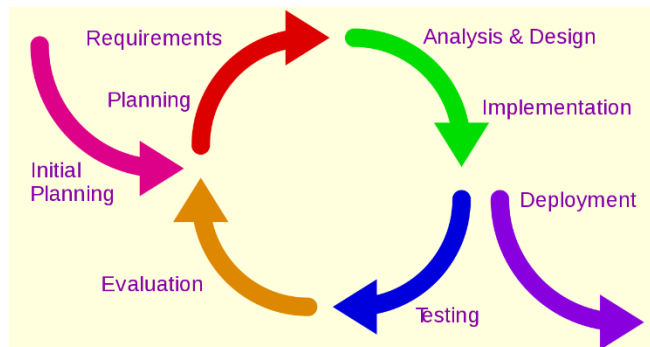


Fig. 2. Metodologia: Desenvolupament iteratiu i incremental.

Per tant, es partirà sobre el projecte base, en el nostre cas és: el corrector amb les etiquetes. El qual serà el nucli, posteriorment s'anirà inclouent la resta de funcionalitats necessàries, com a miniprojectes. Per exemple: poder realitzar login, importar/exportar fitxers. També, per altra banda, el fet d'anar realitzant el projecte en iteracions permet conèixer el progrés real d'aquest i extrapolar si la seva finalització es viable en la data prevista cosa que ens ajudarà si ens trobem amb problemes o desviacions.

## 8 PLANIFICACIÓ

Per una banda hi havia els lliuraments del treball que s'havien de realitzar al campus i per altra la planificació de tasques del projecte.

### 8.1 Lliuraments del treball

La planificació per als lliuraments del treball va ser la següent:

- 07/10: Informe inicial.
- 11/11: Informe de progrés I.
- 23/12: Informe de progrés II.
- 27/01: Proposta informe final.
- 10/02: Proposta de presentació.
- 11/02: Lliurament final al tutor.
- 17/02: Lliurament pòster.
- 23/02: Defensa TFG.

### 8.2 Planificació projecte

Per a la planificació del projecte, primerament hi haurà la part de formació i investigació sobre les eines a utilitzar, fins la primera creació de l'eina de text bàsic. A partir d'allà hi anirem afegint les funcionalitats necessàries i anar incrementant el projecte *Veure Apèndix A1*:

- 15/10: Investigar Javascript amb els diferents plugins i backend.
- 29/10: Investigar backend i iniciar la creació de l'eina.
- 12/11: Crear el sistema login i dissenyar el primer editor de text bàsic.
- 3/12: Importar fitxers Word / OpenOffice i integrar el corrector bàsic.
- 24/12: Creació d'etiquetes i associar-les als documents.

- 14/01: Exportar en document Word / OpenOffice.
- 28/01: Enviar el document i emmagatzemar-lo en la base de dades.

Aquest era el plantejament inicial del projecte però es va modificar per a tenir temps d'introduir millores. Així doncs, es va realitzar un canvi de la planificació i es va adaptar, ja que també van sorgir una sèrie de problemes, els quals s'explicaran en l'apartat d'anàlisi i disseny. Es va haver de refer la planificació *Veure Apèndix A1*, amb el resultat següent:

- 15/10: Investigar Javascript amb els diferents plugins i backend.
- 29/10: Investigar Javascript i plugins i iniciar la creació de l'eina.
- 12/11: Dissenyar i implementar el primer editor de text ric, iniciar la creació d'etiquetes.
- 26/11: Crear etiquetes i associar-les als texts, crear la base de dades i crear el sistema de login.
- 10/12: Emmagatzemar el fitxer modificat a la base de dades i importar fitxers llegibles per l'eina.
- 24/12: Emmagatzemar el fitxer original i agrupar les etiquetes per subcategories.
- 03/01: Modificacions en el disseny d'estructura de la base de dades. Exportar fitxers i enviar-los als alumnes.
- 17/01: Modificacions i millores posteriors.

## 9 DISSENY

En aquest apartat indicarem com s'ha dissenyat en relació a l'estructura, tant de base de dades com del servidor. També s'indicaran els diferents inconvenients i problemes que s'han trobat a l'hora de realitzar aquest disseny i com s'han solucionat.

Com a servidor web, tenim finalment un sistema Apache amb una base de dades MariaDB[16]. Aquest sistema integra un sistema de login mitjançant sessions PHP i connexions a la base de dades, que posteriorment pot ser modificat pel sistema de login de la UAB. També integra un plugin de JavaScript anomenat TinyMCE[9] el qual és un editor de text enriquit, conegut com a WYSIWYG perquè l'edició i visualització del text mantingui el format el més semblant possible del document original i sigui un format que qualsevol usuari pot reconèixer les diferents eines que conté i com utilitzar-les.

### 9.1 Disseny base de dades

El sistema de base de dades està organitzat de la següent manera:

- Primerament tenim una base de dades per a cada any, ja que segons es va comentar es volia una separació entre anys *Veure Apèndix A3*.
- Dintre de la base de dades de cada any tenim les següents taules:
  - Una taula per a cada alumne: On es

guarda un id, el Niu i email de l'alumne.

- Una taula de professors: On es guarda un id, el Niu, contrasenya per accedir a l'eina i l'email del professor.
- Una taula d'assignatures: On es guarda un id, el codi i nom de l'assignatura.
- Per últim, una taula de documents, on es guarda: L'id l'assignatura, l'id de l'alumne, el contingut del document original que s'envia, el contingut del document modificat pel professor, les rutes on s'emmagatzemen aquests documents, l'id del professor que fa la correcció i la data en què s'envia aquesta informació a la base de dades.

### 9.2 Disseny del servidor

Finalment, s'ha decidit implementar el sistema amb Apache, encara que l'idea original era integrar un backend amb Django[10] però van sorgir una sèrie de complicacions, que s'explicaran en l'apartat 9.3. Per tant, tenim un sistema Apache, en el qual es fan les diferents crides PHP, Mysql i Javascript.

Està estructurat en diferents carpetes:

- Css: Carpeta que conté l'estil de la pàgina.
- Js: On es fa la crida al plugin de jquery.
- Php: On tenim tots els fitxers que realitzen les funcions php com ara la comprovació del login o dels fitxers pujats.
- Plugin: Aquí tenim tot el referent amb el plugin que utilitzem per a generar el fitxer de text ric TinyMCE.
- Tmp: On es pujarà primerament el fitxer de l'usuari per a la futura edició.
- Uploads: on es guarden els fitxers pujats tant l'original com el modificat.

### 9.3 Problemes en el disseny

La idea inicial era integrar un backend anomenat Django, ja que el llenguatge que utilitza és Python i és el llenguatge amb el que més es treballa al llarg del grau. Però a l'hora d'integrar el plugin al voltant del qual es realitzaria tot el projecte (TinyMCE) hi havia molta dificultat a l'hora d'integrar-lo, ja que el plugin per a Django era bastant nou, es trobava encara en estat precari i disposava de poca documentació. Per tant, després d'una reunió amb un professor de suport es va decidir prescindir de Django i seguir el projecte únicament amb Apache.

## 10 DESENVOLUPAMENT

Aquest apartat el dividirem en el desenvolupament del backend i del frontend:

### 10.1 Desenvolupament backend

A l'hora de realitzar el desenvolupament es va començar amb la recerca d'algun plugin editor de text ric. Això significa que en un text es conservarà el format original (negretes, text centrat, ...). En canvi, amb una caixa de text normal es veuria com a text pla, sense tenir en compte totes aquestes funcions de format. Es va localitzar el plu-

gin TinyMCE el qual comptava amb molta informació i alguns manuals, també amb diferents subplugins que podien ajudar a la realització de les diverses funcions que es requerien.

Una vegada trobat el plugin es va integrar amb el servidor Apache i es va poder comprovar com el text mantenia el format.

La següent etapa va ser copiar un text i intentar afegir-hi etiquetes. Per a desenvolupar aquesta part, es va utilitzar una funció del plugin per a obtenir una tros de text seleccionat i aquest retornar-lo dintre de les diferents etiquetes que nosaltres havíem dissenyat. També es va integrar un corrector ortogràfic de paraules en anglès.

Una vegada obtingut l'editor de text base, es va procedir a crear el sistema de la base de dades on es guardaria el document, tant original com corregit i on estarien els diferents usuaris que podrien accedir a les funcionalitats del sistema, els professors.

Amb la creació de la base de dades, quedava guardar el document corregit a la base de dades. Es va poder realitzar guardant un format de text pla amb el contingut de la caixa de text del TinyMCE però el problema era que no es podia guardar el contingut original, ja que no hi havia manera de detectar quin era. Per a realitzar-ho vam passar a la següent etapa. La funcionalitat d'obrir documents Word o Odt des del navegador i que aquests es mostressin a la caixa de text. Com s'explica en l'apartat 10.3 es pot veure com es va desestimar aquesta opció i el perquè. Per tant es va buscar una alternativa i es va trobar que amb documents format .html no hi havia cap problema, tampoc era massa difícil pels professors canviar el format i el resultat era bàsicament el mateix que el format original. Per a la conversió de fitxers es pot fer massivament amb moltes eines, per exemple MultiDoc Converter.

Aquests fitxers es van definir amb una nomenclatura per a identificar els alumnes, l'assignatura i el nom de l'exercici. Per fer-ho es va fer amb el format: "CodiAssignatura\_NIU\_NomExercici", per ex: "TEST\_1331964\_P1". Per comprovar que els formats fossin correctes, tant dels noms dels fitxers, com d'extensions i que el login d'usuari fos correcte, s'ha utilitzat Ajax:

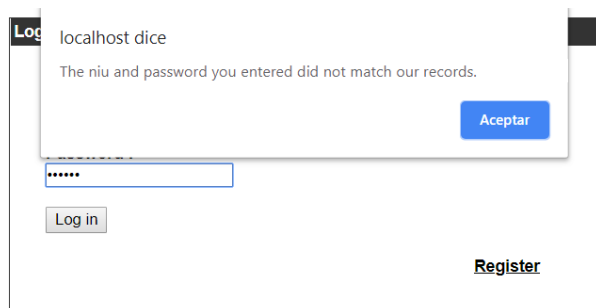


Fig. 3. Exemple de control amb Ajax.

Una vegada convertit el fitxer es puja en una carpeta temporal on posteriorment s'agafarà el contingut i es guardarà a la carpeta Uploads, tant el fitxer corregit com l'original. A la base de dades a part del contingut de la caixa de text, també es guardarà la ubicació dels fitxers, tant l'original com el corregit. Una vegada emmagatzemats,

s'enviarà el fitxer corregit de tornada a l'alumne mitjançant la llibreria PHPMailer[17] i el servidor SMTP escollit.

## 10.2 Desenvolupament frontend

A l'hora de realitzar el desenvolupament del frontend, es va tenir en compte que fos un entorn intuïtiu, ja que els professors que han d'utilitzar l'eina no posseïen molts coneixements d'informàtica. Per tant es va començar amb un sistema de login bàsic i una pujada de fitxers estàndard. En l'apartat de les etiquetes, es va optar per agrupar-les segons el tipus i diferenciar cada tipus amb una gamma de color:

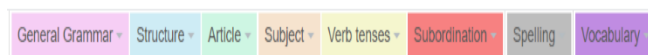


Fig. 4. Etiquetes.

- General Grammar:
  - Saxon Genitive
  - Countable / Uncountable
  - Morphology
  - Wrong pronoun
  - Other problems
- Structure:
  - Too long / complex sentences
  - Incorrect parallel structure
  - Incorrect word order
  - The whole structure is incorrect
- Article:
  - Overuse of article
  - Underuse of article
  - Wrong determiner
- Subject:
  - Missing subject
  - Double subject
  - Subject-verb agreement
- Verb tenses:
  - Wrong conditional
  - Verb tense misuse
- Subordination:
  - Problems with relative clauses
  - Incorrect subordination
- Spelling:
  - Problem with spelling
- Vocabulary:
  - Problema with vocabulary

Així el professor trobarà de manera més visual els diferents errors i podrà veure en el text els tipus d'errors que hi ha actualment. També s'ha afegit un apartat de comentaris, ja que era un requeriment per als professors, perquè a part d'indicar els errors, també havien de poder afegir els comentaris que consideressin oportuns. Per a que poguessin afegir aquests comentaris, s'ha afegit una caixa de text amb el voltant vermell sota el títol de Comments, on els professors poden escriure els comentaris globals respecte a la descripció de l'alumne.

### 10.3 Problemes desenvolupament

A l'hora de realitzar el desenvolupament el principal problema va ser com obrir fitxers en format Word o Odt des de l'eina web. Després de moltes proves i consultes, no es va veure viable. La professora d'anglès ens va indicar que es podrien convertir els documents i així ens va ser possible trobar una solució. Es va provar amb format rtf, pdf i uns quants més fins que finalment amb el format html sembla que el document original i el que es mostrava per l'editor eren pràcticament iguals. En ser redaccions el que interessava era mantenir el format del text. Així doncs es va procedir amb el format html.

Per altra banda, a l'hora d'enviar correus, es va trobar que l'enviament per defecte no deixava adjuntar fitxers, per tant es va haver de buscar alguna opció per a poder enviar els correus, es va trobar l'opció de la llibreria PHPMailer però necessitava un correu SMTP per a funcionar correctament i després de provar uns quants servidors, es va trobar que amb Mailjet funcionava correctament.

### 10.4 Tests realitzats

Els tests que es van realitzar els podem diferenciar en dues parts: els tests del backend i els tests del frontend.

En l'apartat de backend podem destacar:

- En el sistema de login es va comprovar que no es pogués fer mysql injection amb les funcions: stripslashes i mysqli\_real\_escape\_string. També si el password o l'usuari estava buit que retornes error.
- Es va encriptar la contrasenya de l'usuari amb hash i es va comprovar que anava guardant contrasenys diferents.
- Per altra banda també en l'apartat de fitxers, que només es poguessin pujar fitxers htm o html.
- Es va provar també que el format de l'arxiu fos el correcte amb la funció regex de javascript.
- A part, també es verifica que el codi de l'assignatura existeixi i el niu de l'alumne tingui el format correcte.

En l'apartat de frontend trobem:

- Una vegada es pujava un fitxer a la carpeta temporal, s'analitzava si tenien un encode diferent del utf-8, en cas afirmatiu, s'havia de convertir perquè sinó es mostrava el text amb caràcters incorrectes.
- També es va provar d'entrar a les pàgines de pujar fitxer o a la del corrector sense esta registrat i et retornava a la pàgina de registre.
- A l'hora d'entrar a pàgines d'administració sense ser administrador, retornaven a la pàgina de pujada de fitxer.

## 11 EXPOSICIÓ DE RESULTATS

En aquest apartat s'indicarà els diferents procediments que tenen els professors per a realitzar la correcció de documents visualment.

Primerament tenim l'apartat de login, es comprovarà amb

la taula de professors si existeix l'usuari i si coincideix la contrasenya.

Una vegada realitzat el login, arribarem a l'apartat de pujada de fitxer. Tindrem tres opcions: la pujada de fitxer, el canvi de contrasenya i l'apartat de tancar sessió.

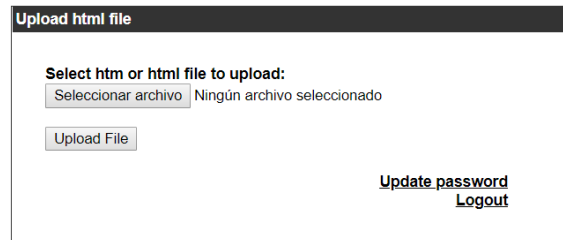


Fig. 5. Menú de pujada de fitxer.

L'opció de tancar sessió retorna a la pàgina d'inici, per tant, veurem primerament la funció d'actualitzar contrasenya:

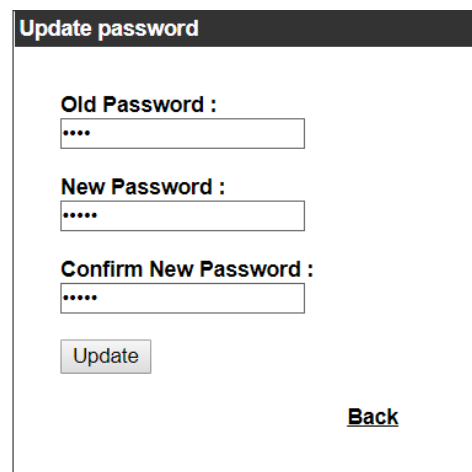


Fig. 6. Actualització de contrasenya.

La següent opció es la de pujar el fitxer fer-ho seleccionarem el fitxer html i el pujarem mitjançant el gestor següent:

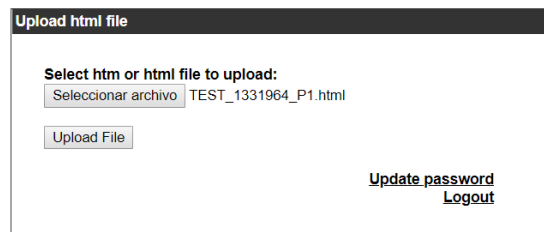


Fig. 7. Fitxer seleccionat per a pujar.

En aquest cas de mostra s'ha indicat un fitxer amb el codi de l'assignatura TEST, el niu de l'estudiant 1331964, i el nom de l'exercici P1.

Ara el professor passarà a l'àrea de correcció on podrà marcar diferents errors indicant el tipus. Tindrà un menú desplegable agrupat per categories i subcategories.

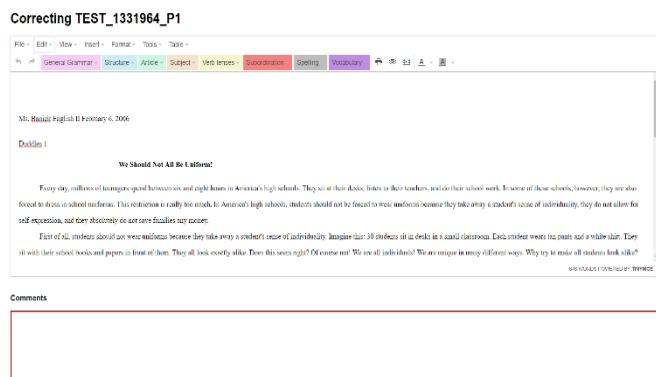


Fig. 8. Vista general eina de correcció.

Seleccionarà un conjunt de text i posteriorment indicarà el tipus d'error. *Veure Apèndix A4.*

En el cas que es trobi més d'un error a una mateixa frase, aquest podrà ser indicat sense problemes, ja que l'error més global contindrà el més concret. *Veure Apèndix A5.*

El professor podrà afegir comentaris en la caixa de text següent, la qual es visualitzarà com un requadre vermell amb el text de comentaris a l'interior:



Fig. 9. Apartat de comentaris.

Finalment el professor enviarà el document perquè s'emmagatzemi a la base de dades tant l'original com el corregit. Es pot observar les diferències entre els dos documents resultants en *Apèndix A6* i *Apèndix A7*.

En cas de que el document contingui un correu a l'interior s'enviarà directament al correu de l'alumne. En cas contrari, es podrà introduir un nou correu on enviar aquest document. Una vegada enviat el correu, tant l'alumne com el professor que envia el treball el rebran com en la següent imatge:

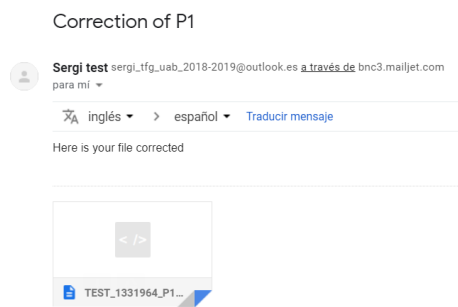


Fig. 10. Correu amb el fitxer corregit.

Si l'usuari és l'administrador, no podrà corregir treballs però podrà crear professors i assignatures:

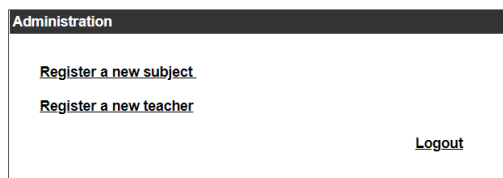


Fig. 11. Panell d'administració.

Si registra una nova assignatura, se li obrirà una nova pàgina on haurà d'introduir el codi i nom de l'assignatura:

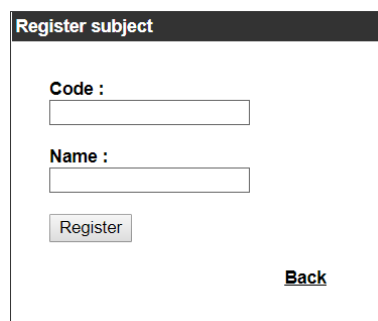


Fig. 12. Registrar assignatura.

Si registra un nou professor, se li obrirà una nova pàgina on haurà d'introduir el niu, la contrasenya i el correu del professor:

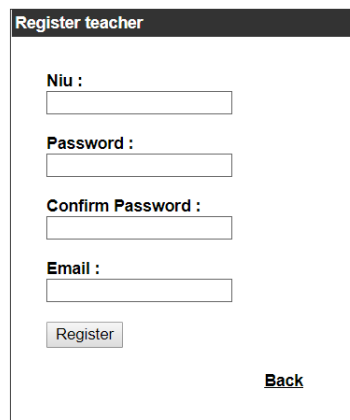


Fig. 13. Registrar professor.

## 12 CONCLUSIONS

Les conclusions del projecte s'han realitzat a partir dels objectius proposats i els resultats obtinguts.

Com a objectiu principal s'havia proposat la creació d'una eina que permetés als professors corregir treballs d'anglès de manera estandard. Com a resultat, es pot observar com s'ha desenvolupat correctament aquest apartat i s'ha pogut realitzar una eina intuïtiva. De manera simplificada resol els problemes d'organització, tenint un conjunt d'etiquetes per a marcar cada tipus d'error de l'alumne. També es demanava un corrector que indiqués les paraules mal escrites, cosa que s'ha incorporat a l'eina facilitant la feina dels professors.



Per acabar, podem destacar un altre objectiu molt important que consistia en que el document pogués ser enviat de tornada a l'alumne de manera automatitzada. Per aconseguir aquest objectiu, els professors només necessitaran fer clic al botó de "Send" i l'alumne tindrà al seu correu el document corregit amb els comentaris, cosa que, tant pels alumnes com pels professors els hi suposa una millora considerable respecte a com es gestionava anteriorment.

### 13 LÍNIES FUTURES

Com a línies de continuació de projecte es podria obtenir per una ampliació de les etiquetes amb noves consideracions que els professors puguin indicar. Extrapolant aquest corrector a altres idiomes, vist que el funcionament de correcció és força similar. També, es podria desenvolupar un cercador per alumnes o assignatures o per tipus d'error on els professors puguessin veure les redaccions corresponents. Per altra banda, tenint una base de dades d'errors ortogràfics i de vocabulari d'alumnes, es podria desenvolupar un corpus d'errors més comuns on desenvolupar estadístiques i tipus d'errors més comuns per així ajudar a corregir-los enfocant-se en reforçar exercicis per evitar aquest tipus d'errors.

### AGRAÏMENTS

Els agraïments del projecte van dirigits en primer lloc a la meua tutora Marta Prim, el seu suport i la seva experiència que m'han ajudat a enfocar i guiar el projecte durant el transcurs d'aquest. En segon lloc a la professora Ana Fernández, la qual va proposar aquest projecte pel seu constant feedback i la seva adaptació. En tercer lloc al professor Vicenç Soler per l'ajuda proporcionada i per ajudar-me a enfocar correctament el projecte en uns moments de dubte. I per acabar però no menys important a la família i amics que m'han donat suport al llarg del transcurs del Grau.

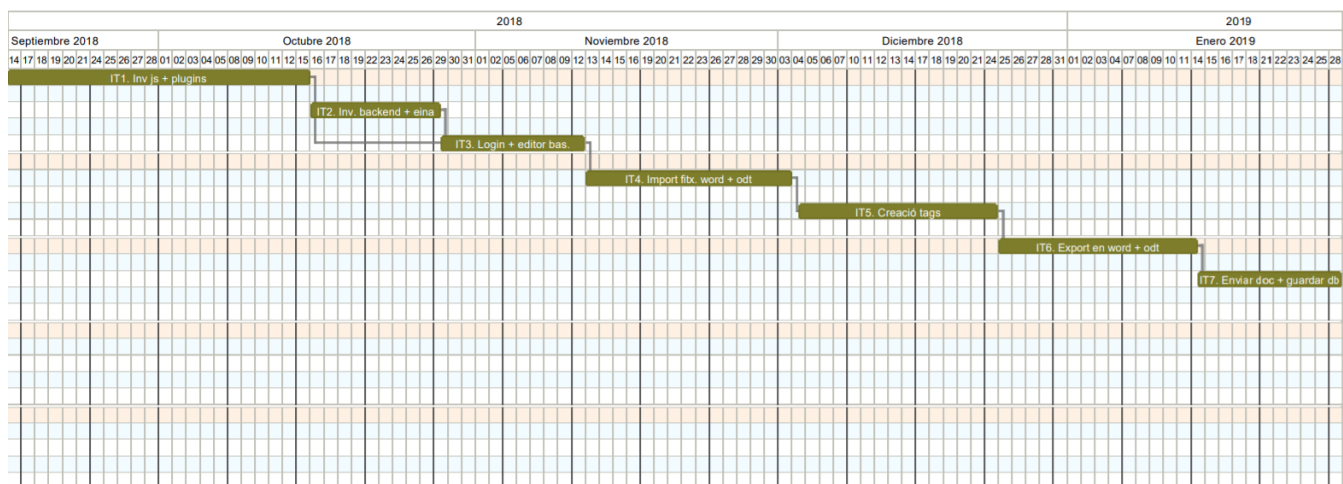
### BIBLIOGRAFIA

- [1] Grammarly [En línia]. Disponible a: <https://www.grammarly.com/> Consultat el 29 de setembre de 2018.
- [2] Reverso [En línia]. Disponible a: [http://www.reverso.net/text\\_translation.aspx?lang=ES](http://www.reverso.net/text_translation.aspx?lang=ES) Consultat el 29 de setembre de 2018.
- [3] Online corrector [En línia]. Disponible a: <https://www.onlinecorrection.com/> Consultat el 29 de setembre de 2018.
- [4] Microsoft Word [En línia]. Disponible a: <https://products.office.com/es-es/word> Consultat el 29 de setembre de 2018.
- [5] LibreOffice Writer [En línia]. Disponible a: <https://es.libreoffice.org/descubre/writer/> Consultat el 29 de setembre de 2018.
- [6] Microsoft Visual Studio "Word JavaScript add-in APIs" [En línia]. Disponible a: <https://www.youtube.com/watch?v=EaLVfaj1mW8> Consultat el 30 de setembre de 2018.
- [7] XcriptMonster "1. Creating a Rich Text Editor WYSIWYG Using HTML and JavaScript - Introduction, adding HTML" [En línia]. Disponible a: [https://www.youtube.com/watch?v=S4K\\_PoWTCz4](https://www.youtube.com/watch?v=S4K_PoWTCz4) Consultat el 30 de setembre de 2018.
- [8] Proyectos ágiles [En línia]. Disponible a: <https://proyectosagiles.org/desarrollo-iterativo-incremental/> Consultat el 06 d'octubre de 2018.
- [9] TinyMCE "Build Beautiful Content for the Web with Tiny" [En línia]. Disponible a: <https://www.tiny.cloud/> Consultat el 10 d'octubre de 2018.
- [10] TinyMCE integration for Django [En línia]. Disponible a: <https://django-tinymce.readthedocs.io/en/latest/> Consultat el 28 d'octubre de 2018.
- [11] How to Set up TinyMCE in Django [En línia]. Disponible a: <https://fosstack.com/how-to-set-up-tinymce-in-django-app/> Consultat el 28 d'octubre de 2018.
- [12] HackedExistence, "Django Tutorial 4 – TinyMCE" [En línia]. Disponible a: <https://www.youtube.com/watch?v=bJeTEDRvGVA> Consultat el 28 d'octubre de 2018.
- [13] Yusuf Shakeel, "How to setup TinyMCE | TinyMCE #01" [En línia]. Disponible a: <https://www.youtube.com/watch?v=8hNvH3R-haY> Consultat el 3 de novembre de 2018.
- [14] Stackoverflow, "How do I render a Word document (.doc, .docx) in the browser using JavaScript?" [En línia]. Disponible a: <https://stackoverflow.com/questions/27957766/how-do-i-render-a-word-document-doc-docx-in-the-browser-using-javascript> Consultat el 12 de novembre de 2018.
- [15] Editor Appearance [En línia]. Disponible a: <https://www.tiny.cloud/docs/configure/editor-appearance/> Consultat el 14 de novembre de 2018.
- [16] The MariaDB Foundation [En línia]. Disponible a: <https://mariadb.org/> Consultat el 20 de novembre de 2018.
- [17] Tutorial Republic, "PHP MySQL Login System" [En línia]. Disponible a: <https://www.tutorialrepublic.com/php-tutorial/php-mysql-login-system.php> Consultat el 7 de desembre de 2018.
- [18] W3schools, "PHP Select Data From MySQL" [En línia]. Disponible a: [https://www.w3schools.com/php/php\\_mysql\\_select.asp](https://www.w3schools.com/php/php_mysql_select.asp) Consultat el 7 de desembre de 2018.
- [19] Stackoverflow, "Submitting HTML form using JQuery AJAX" [En línia]. Disponible a: <https://stackoverflow.com/questions/16323360/submitting-html-form-using-jquery-ajax> Consultat el 14 de desembre del 2018.
- [20] SMTP, "Qué es un servidor SMTP" [En línia]. Disponible a: <https://serversmtp.com/es/que-es-servidor-smtp/> Consultat el 21 de desembre de 2018.
- [21] Mandrill, "How to send with PHPMailer" [En línia]. Disponible a: <https://mandrill.zendesk.com/hc/en-us/articles/205582147-How-to-Send-with-PHPMailer> Consultat el 3 de gener del 2019.
- [22] Mailjet, Servicio de servidor SMTP [En línia]. Disponible a: <https://es.mailjet.com/feature/smtp-relay/> Consultat el 7 de gener de 2019.
- [23] Bitbucket [En línia]. Disponible a: <https://bitbucket.org> Consultat el 10 de gener de 2019.

## APÈNDIX

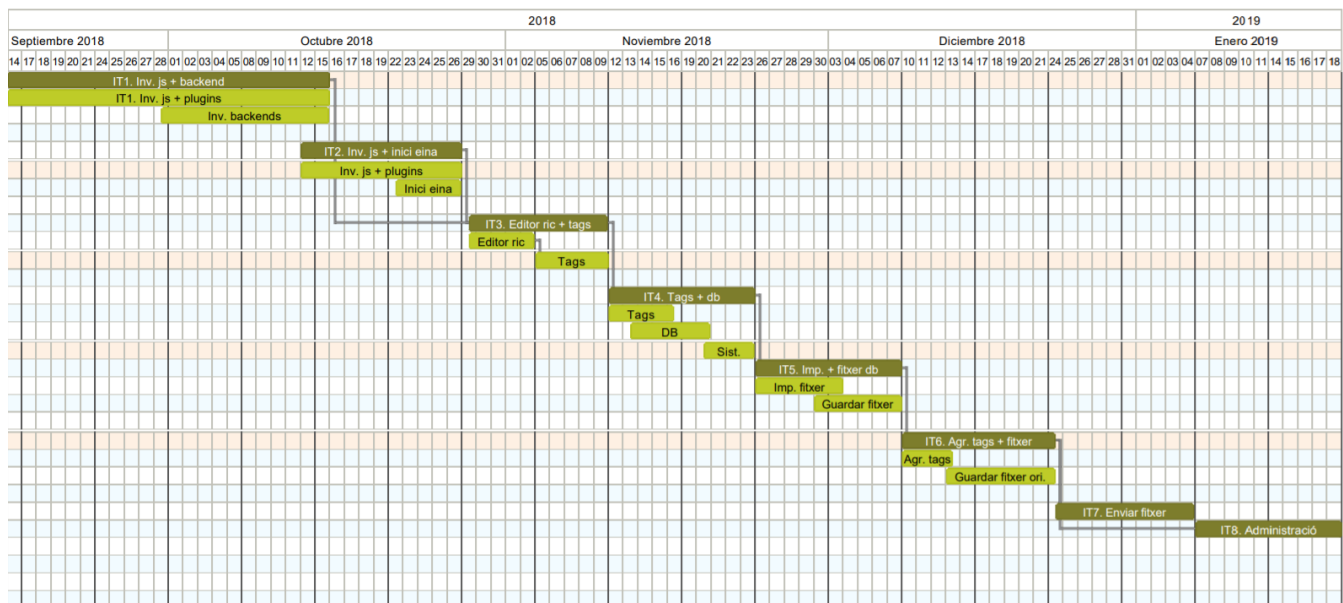
### A1. PLANIFICACIÓ DEL PROJECTE. DIAGRAMA DE GANTT ORIGINAL.

Aquí es pot observar el diagrama de Gantt original del projecte:



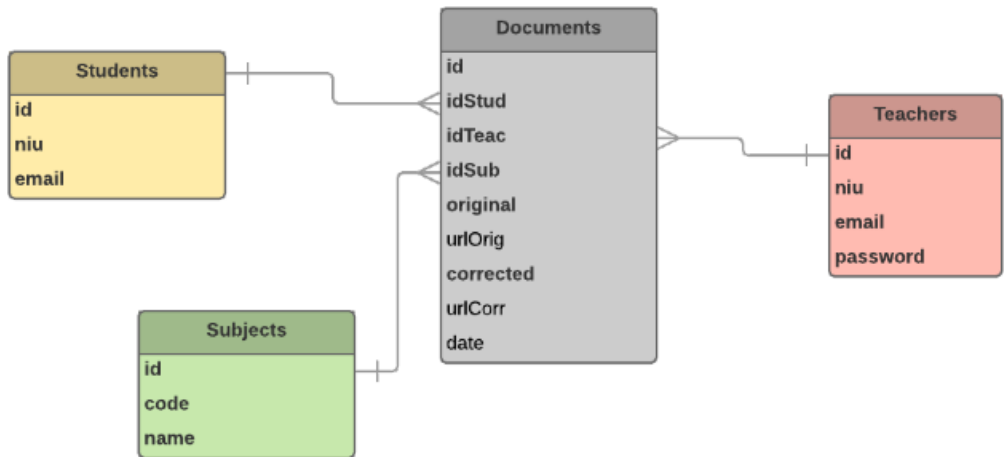
### A2. PLANIFICACIÓ DEL PROJECTE. DIAGRAMA DE GANTT FINAL.

Aquí es pot observar el diagrama de Gantt resultant dels inconvenients trobats i de les noves iteracions del projecte:



### A3. DIAGRAMA E-R BASE DE DADES. ANY 2019

Per a cada any es crea una base de dades amb les mateixes característiques:



### A4. Agrupació categories d'etiquetes.

#### Correcting TEST\_1331964\_P1

File Edit View Insert Format Tools Table

↶ ↷
General Grammar
Structure
Article
Subject
Verb tenses
Subordination
Spelling
Vocabulary
🖨 👁 🔍 A A

Ms. Bar... 006

Duddles

**We Should Not All Be Uniform!**

Every day, millions of teenagers spend between six and eight hours in America's high schools. They sit at their desks, listen to their teachers, and do their school work forced to dress in school uniforms. This restriction is really too much. In America's high schools, students should not be forced to wear uniforms because they take away a st self-expression, and they absolutely do not save families any money.

First of all, students should not wear uniforms because they take away a student's sense of individuality. Imagine this: 30 students sit in desks in a small classroom. Ea sit with their school books and papers in front of them. They all look exactly alike. Does this seem right? Of course not! We are all individuals! We are unique in many differ This idea is supported by testimony from a 17-year-old student forced to wear uniforms. Her story was reported in *National Catholic Reporter* of March, 2002, and she said,

## A5. ETIQUETA ENGLOBANT DUES ETIQUETES MÉS.

### Correcting TEST\_1331964\_P1

Ms. Banick English II February 6, 2006

Duddles 1

**We Should Not All Be Uniform!**

[G-LCS: Too long / complex sentences]Every day, millions[G-C/U: Countable/uncountable] of teenagers[/G-C/U] spend [G-SG: Saxon genitive]between six and eight hours[/G-SG] in America's high schools[/G-LCS]. They sit at their desks, listen to their teachers, and do their school work. In [G-SP: Spelling]some[/G-SP] of these schools, however, they are also forced to dress in school uniforms. This restriction is really too much. In America's high schools, students [G-VT: Verb tense misuse]should not be forced[/G-VT] to wear uniforms because they take away a student's sense of individuality, they do not allow for self-expression, and they absolutely do not save families any money.

First of all, students should not wear uniforms because they take away a [G-UnA: Underuse of article]student's[/G-UnA] sense of individuality. Imagine this: 30[G-RC: Problems with relative clauses] students sit in desks in a small classroom. E[/G-RC]ach student wears tan pants and a white shirt. They sit with their school books and papers in front of them. They all look exactly alike. Does this seem right? Of course not! We are all

## A6. FITXER ORIGINAL EMMAGATZEMAT A LA BASE DE DADES.

Ms. Banick English II February 6, 2006

Duddles 1

**We Should Not All Be Uniform!**

Every day, millions of teenagers spend between six and eight hours in America's high schools. They sit at their desks, listen to their teachers, and do their school work. In some of these schools, however, they are also forced to dress in school uniforms. This restriction is really too much. In America's high schools, students should not be forced to wear uniforms because they take away a student's sense of individuality, they do not allow for self-expression, and they absolutely do not save families any money.

First of all, students should not wear uniforms because they take away a student's sense of individuality. Imagine this: 30 students sit in desks in a small classroom. Each student wears tan pants and a white shirt. They sit with their school books and papers in front of them. They all look exactly alike. Does this seem right? Of course not! We are all individuals! We are unique in many different ways. Why try to make all students look alike? This idea is supported by testimony from a 17-year-old student forced to wear uniforms. Her story was reported in *National Catholic Reporter* of March, 2002, and she said, "Everyone hated it. It completely killed any sense of individuality any one of us had. Everyone looked the same. It was sad to

watch" (3). Clearly, students with experience in wearing uniforms feel that they take away a sense of individuality.

Next, school uniforms do not allow students to express themselves. Teenagers are at an age where they are trying to establish their identity. Is it fair to prevent them from exploring self-expression by making them wear uniforms? By taking away a student's right to choose what to wear, we also take away the opportunity to express oneself. For example, if a young woman is very interested in becoming a fashion designer, she may begin to take home economics classes and train herself to make clothes and create patterns. If that were true, she would definitely want to wear her own creations to school; this would allow her to show off her achievements and gauge her peers' responses to her newly created clothes. However, if she were a student who was forced to wear uniforms, she would not be able to express herself in this manner. To take that right away from America's youth is an injustice.

Finally, some schools tell parents that uniforms are a good idea because they save families money. This is not necessarily true. It is logical to say that when teenagers come home from school, they want to change out of the clothes that they were forced to wear to school.

Well, what are they going to change into? They will want clothes of their own choice, of course. This means that parents would still have to provide their children with clothes that fit current trends, *plus* they would have to buy a separate set of clothes for their children to wear to school each day. While it is possible that some teenagers may not want designer clothes in addition to their school uniforms, it is very unlikely.

In conclusion, in our public schools, students should not be forced to wear uniforms. This is because they take away a student's sense of individuality, they do not allow for self-expression, and they do not save families any

## A7. FITXER MODIFICAT EMMAGATZEMAT A LA BASE DE DADES.

Ms. Banick English II February 6, 2006

Duddles 1

We Should Not All Be Uniform!

**[G-LCS: Too long / complex sentences]**Every day, millions **[G-C/U: Countable/uncountable]** of teenagers **[G-C/U]** spend **[G-SG: Saxon genitive]** between six and eight hours **[G-SG]** in America's high schools **[G-LCS]**. They sit at their desks, listen to their teachers, and do their school work. In **[G-SP: Spelling]** some **[G-SP]** of these schools, however, they are also forced to dress in school uniforms. This restriction is really too much. In America's high schools, students **[G-VT: Verb tense misuse]** should not be forced **[G-VT]** to wear uniforms because they take away a student's sense of individuality, they do not allow for self-expression, and they absolutely do not save families any money.

First of all, students should not wear uniforms because they take away a **[G-UnA: Underuse of article]** student's **[G-UnA]** sense of individuality. Imagine this: 30 **[G-RC: Problems with relative clauses]** students sit in desks in a small classroom. **E** **[G-RC]** each student wears tan pants and a white shirt. They sit with their school books and papers in front of them. They all look exactly alike. Does this seem right? Of course not! We are all individuals! We are unique in many different ways. Why try to make all students look alike? This idea is supported by testimony from a 17-year-old student forced to wear uniforms. Her story was reported in National Catholic Reporter of March, 2002, and she said, "Everyone hated it. It completely killed any sense of individuality any one of us had. Everyone looked the same. It was sad to

watch" (3). Clearly, students with experience in wearing uniforms feel that they take away a sense of individuality.

Next, school uniforms do not allow students to express themselves. Teenagers are at an age where they are trying to establish their identity. Is it fair to prevent them from exploring self-expression by making them wear uniforms? By taking away a student's right to choose what to wear, we also take away the opportunity to express oneself. For example, if a young

woman is very interested in becoming a fashion designer, she may begin to take home economics classes and train herself to make clothes and create patterns. If that were true, she would definitely want to wear her own creations to school; this would allow her to show off her achievements and gauge her peers' responses to her newly created clothes. However, if she were a student who was forced to wear uniforms, she would not be able to express herself in this manner. To take that right away from America's youth is an injustice.

Finally, some schools tell parents that uniforms are a good idea because they save families money. This is not necessarily true. It is logical to say that when teenagers come home from school, they want to change out of the clothes that they were forced to wear to school.

Well, what are they going to change into? They will want clothes of their own choice, of course. This means that parents would still have to provide their children with clothes that fit current trends, plus they would have to buy a separate set of clothes for their children to wear to school each day. While it is possible that some teenagers may not want designer clothes in addition to their school uniforms, it is very unlikely.

---