



GRAU EN ENGINYERIA INFORMÀTICA

TREBALL DE FINAL DE GRAU

---

Implementació d'un programa de neteja  
d'informes per al buscador semàntic  
d'ActualPacsv4

---

*Autor:*  
Mario ARROYO TENA

*Supervisor:*  
Rafael FORCADA MARTINEZ  
*Tutor acadèmic:*  
Angel Pascual DEL POBIL FERRÉ

Data de lectura: 15 de Juliol de 2020  
Curs acadèmic 2019/2020

## Resum

Aquest projecte s'ha desenvolupat seguint una metodologia de desenvolupament del programari tradicional guiada per el PMBOK. Ha estat realitzat al llarg dels 3 mesos de pràctiques. Un dels tres mesos en l'empresa Actualtec S.L., ubicada en Espaitec 2 i els altres dos mesos s'han realitzat des de casa a causa de la COVID-19. Dins de l'abast del projecte s'han realitzat totes les fases de la metodologia tradicional (Inici, Planificació, Gestió, Anàlisi, Disseny, Implementació, Testeig, Tancament i posada en marxa i Seguiment i control). L'abast del projecte abasta totes les fases menys la del disseny, ja que, el programa ja es troba implementat per l'empresa i allò que es desitjava era ampliar la seva funcionalitat. No obstant, el disseny s'utilitzarà per a mostrar aquesta ampliació en la demostració. L'objectiu del producte desenvolupat és recollir estudis, dels quals es puguen obtindre informes radiològics i netejar-los d'informació sensible. Després, l'empresa enviarà un arxiu comprimit amb els informes nets a Semanticbots, per a que els indexen en l'API del buscador semàntic. Per a concloure, el projecte ha finalitzat correctament i, durant el desenvolupament, no he parat d'aprendre noves tecnologies i tècniques que s'utilitzen en el món laboral del desenvolupament del programari.

## Paraules Clau

- Python
- Radiologia
- Informes
- Buscador semàntic

## Keywords

- Python
- *Radiology*
- *Reports*
- *Semantic searcher*

# Índex

<b>1</b>	<b>Introducció</b>	<b>7</b>
1.1	Context i motivació del projecte . . . . .	7
1.2	Objectius del projecte . . . . .	8
1.3	Descripció del projecte . . . . .	9
1.4	Estructura de la memòria . . . . .	9
<b>2</b>	<b>Planificació del projecte</b>	<b>11</b>
2.1	Metodologia . . . . .	11
2.2	Planificació . . . . .	11
2.3	Gestió . . . . .	11
2.3.1	Gestió de l'abast . . . . .	12
2.3.2	Gestió del temps . . . . .	22
2.4	Seguiment del projecte . . . . .	25
<b>3</b>	<b>Anàlisi i disseny del sistema</b>	<b>27</b>
3.1	Anàlisi del sistema . . . . .	27
3.1.1	Requisits . . . . .	27
3.1.2	Diagrama de casos d'ús . . . . .	28
3.1.3	Històries d'usuari . . . . .	33

3.2	Disseny de l'arquitectura del sistema . . . . .	33
3.3	Disseny de la interfície . . . . .	34
<b>4</b>	<b>Implementació i proves</b>	<b>37</b>
4.1	Detalls d'implementació . . . . .	37
4.1.1	Mòdul Estudis . . . . .	37
4.1.2	Mòdul Informes . . . . .	38
4.1.3	Mòdul Neteja . . . . .	38
4.1.4	Mòdul Càrrega . . . . .	40
4.1.5	Fitxers .log . . . . .	41
4.1.6	Frontend . . . . .	43
4.2	Verificació y validació . . . . .	45
4.2.1	Test d'integració . . . . .	45
4.2.2	Tests unitaris . . . . .	47
<b>5</b>	<b>Conclusions</b>	<b>53</b>
	<b>Bibliografia</b>	<b>53</b>
<b>A</b>	<b>Manual d'usuari</b>	<b>57</b>
A.0.1	Requisits . . . . .	57
A.0.2	Instal·lació . . . . .	57
A.0.3	Fitxer de configuració config_default.py . . . . .	58
A.0.4	Fitxer words_file_clean.txt . . . . .	60
A.0.5	Ordre d'execució . . . . .	61
	<b>Índex de figures</b>	<b>68</b>

*ÍNDEX*

5

**Índex de taules**

**69**



# Capítol 1

## Introducció

### 1.1 Context i motivació del projecte

El projecte, la memòria final del qual es presenta en aquest document, es va desenvolupar a l'empresa Actualtec S.L. [1], localitzada en Espaitec 2. Aquest és el nom jurídic de l'empresa però també es pot referir a aquesta amb el nom d'ActualMed [2]. Aquesta empresa treballa en l'àmbit de la salut i es dedica majoritàriament a oferir servicis informàtics a radiòlegs i clíniques radiològiques.

El seu producte, enfocat a aquests servicis, és ActualPacs, què ofereix una plataforma on radiòlegs poden emetre i consultar informes més ràpidament que amb les tecnologies tradicionals. Açò redueix les llistes d'espera i augmenta la productivitat del radiòleg.

Aquesta empresa està treballant en diversos projectes actualment, un del quals és ActualPacsv4, la quarta versió del producte original ActualPacs. Aquesta quarta versió va sorgir com a solució al problema del rendiment de l'anterior versió del producte en algun dels seus clients. En ActualPacsv4 s'implementarà un sistema d'ajuda al radiòleg basat en algorismes de *Machine Learning* que aprofiten la gran base de dades per a retroalimentar al metge.

Abans d'explicar el funcionament i objectiu del projecte que s'exposa, cal explicar en que consisteix el programa amb el qual aquest projecte millorarà el resultat. Aquest programa és el buscador semàntic de ActualPacsv4.

El buscador semàntic permet als radiòlegs buscar informació sobre termes en informes i/o articles de la seua clínica radiològica o d'una altra. Aquests informes contenen un identificador, una descripció, una modalitat i la informació del informe. Aquesta informació mostra tècniques i interpretacions de radiologies realitzades a pacients. També mostra el nom dels doctors i dels pacients, així com altres dades personals. Açò últim no és desitjable que aparega en el resultat de una cerca, i per tal que no aparega s'ha de realitzar una neteja dels informes que es mostraran. Aquest buscador semàntic utilitza una API (*Application Programming Interface*) propietat de l'empresa Semanticbots [3]. En aquest projecte es va utilitzar aquesta API per a poder buscar i pujar informes.

Respecte a l'objectiu del projecte que s'ha realitzat durant l'estada en pràctiques, consisteix en la implementació d'un *script* que extraga els informes escrits per metges radiòlegs i després aplicar-los una neteja de dades sensibles. Aquesta neteja deu ser aplicable a qualsevol clínica radiològica o radiòleg client d'ActualMed. Després el buscador semàntic dins d'ActualPacsv4 consumirà aquests informes i els mostrarà als usuaris que han realitzat una cerca.

## 1.2 Objectius del projecte

El principal objectiu d'aquest projecte és proporcionar accés als informes sense dades personals dels pacients i dels doctors. Aquest accés es realitzarà a través del buscador, que facilita la cerca d'informes i articles, ja que es realitza una cerca a partir de la paraula o paraules passades a aquest.

L'objectiu principal és pot desglossar en els següents subobjectius:

- Oferir ajuda en línia als radiòlegs.
- Oferir accés a informes i articles nets de dades personals.
- Agilitzar el procés de cerca d'estudis, informes i articles.
- Organitzar els informes i estudis per any.
- Ajudar a implementar un sistema d'ajuda en línia per a radiòlegs i metges.

L'abast del projecte inclou la part d'implementació del programa que recollirà tots els estudis, els classificarà per any, recollirà tots els informes d'aquests estudis i els netejarà d'informació sensible. Després de netejar els informes s'enviaran i carregaran a un servei de recerca semàntica que s'encarregarà de tornar els resultats a la plataforma. Finalment es realitzarà una interfície d'usuari per a realitzar una demostració més visual en la presentació final d'aquest treball de final de grau.

Per una altra part, l'abast no inclou el disseny ni implementació de la base de dades ni del buscador semàntic, ja que ja està implementat i en funcionament. La base de dades que s'utilitzarà per a realitzar les proves serà una còpia de la base de dades real en un servidor Apache local.

Pel que fa a l'abast informàtic, el primer mes de practiques es va utilitzar l'equip hardware que proporciona l'empresa, un ordinador de sobretaula. I a causa de la COVID-19, la resta de les pràctiques es van realitzar amb el meu ordinador portàtil personal.

Amb aquest equip es desenvoluparà tota la funcionalitat de recollir estudis i informes, i posteriorment la neteja d'aquests informes. No és necessari cap equip específic de *hardware*.

Quant a l'abast organitzatiu, durant les pràctiques i el desenvolupament del projecte, he estat en contacte amb l'equip de programadors de l'empresa. Com el meu treball és una funcionalitat d'una aplicació de l'empresa ja implementada, és necessari treball en equip, per tant, aquest projecte no s'ha realitzat totalment en solitari.



## 1.3 Descripció del projecte

Per al desenvolupament funcional d'aquest programa s'utilitzarà Python com a llenguatge de programació. Per a poder tindre una base de dades de prova en local s'utilitzarà MySQL en un servidor Apache, creat per l'aplicació Xampp [4].

Per al desenvolupament de la demostració tècnica o *Frontend* en la panificació del projecte, s'ha utilitzat Django [5], i Ajax per a la funcionalitat. Gràcies a Ajax es van poder crear *Datatables* per a mostrar les dades de forma correcta i dinàmica, a més va ser fàcil d'integrar amb les vistes. Per crear les vistes, s'ha utilitzat HTML (*HyperText Markup Language*) i Bootstrap ja que s'integren molt bé amb Django. HTML és un llenguatge de marcat i Bootstrap una llibreria molt coneguda per al disseny de pàgines web. Van ser la primera elecció ja que les he utilitzat en assignatures del grau.

L'editor de codi font o IDE utilitzat és PyCharm, ja que és un editor dedicat al llenguatge Python i amb el que he tingut contacte en assignatures del grau.

A part de les aplicacions de programació, també s'han utilitzat altres ferramentes com són: GIT [6] per al control de versions del codi font; Trello [7], per a l'administració i control del projecte; MS Project [8], un programa per a la gestió de projectes; Lucidchart [9], una web per a crear diagrames; i Redmine [10], per a la comunicació i organització amb l'equip de l'empresa.

El projecte intentarà millorar la cerca semàntica d'informes radiològics per a tots els clients d'ActualMed, ja que una vegada implementat el codi, oferirà informes sense informació personal o sensible.

## 1.4 Estructura de la memòria

En aquest apartat s'explica l'estructura d'aquest document, que és la següent:

- Al capítol 2 es mostra la planificació inicial del projecte. En aquest apartat s'explica la metodologia que s'ha seguit per a realitzar el projecte i la planificació. També es mostra l'estimació de recursos i costos de la realització del projecte. Per finalitzar s'explica la gestió que s'ha dut a terme; en concret gestió de l'abast i del temps i el seguiment i control del desenvolupament del projecte.
- Al capítol 3 es descriu la fase d'anàlisi i disseny del sistema realitzat. Dins de l'anàlisi s'expliquen els requisits del programa i les històries d'usuari, i de forma més visual es mostra un diagrama de casos d'ús. En la part de disseny s'explica el desenvolupament i creació dels dissenys de les interfícies realitzades.
- Al capítol 4 es mostra la fase d'implementació i proves. En aquest capítol es detalla el treball de programació realitzat durant les pràctiques per al desenvolupament d'aquest projecte. També s'explica i es mostra el codi de proves que s'han realitzat, en concret proves d'unitat, d'integració i de interfície d'usuari.

- Al capítol 5 inclou les conclusions finals de la realització d'aquest projecte. Es mostren les conclusions tant de l'àmbit formatiu, del professional i del personal.
- Finalment al Annex A es troba la fase de tancament del projecte, l'última fase del desenvolupament del projecte de pràctiques. En aquesta fase es va realitzar, per a l'empresa, un manual d'usuari dels programes realitzats.

## Capítol 2

# Planificació del projecte

### 2.1 Metodologia

La realització de la planificació d'aquest projecte es va dur a terme mitjançant una metodologia predictiva seguint el manual PMBOK, o *A Guide to the Project Management Body of Knowledge*. S'ha decidit seguir una metodologia predictiva ja que els requisits del sistema estaven clars des d'un principi i eren immutables. També perquè el llenguatge de programació i l'entorn de codi font eren coneguts i els havia utilitzat anteriorment en el grau d'Enginyeria Informàtica. Al seguir aquesta metodologia en cascada, a l'acabar una fase no es pot tornar a l'anterior, però durant tot el desenvolupament del projecte també es realitzarà un seguiment i control d'aquestes fases.

### 2.2 Planificació

La planificació ha sigut realitzada seguint una metodologia predictiva, com ja s'ha explicat abans, en concret seguint el cicle de vida del programari en cascada. Les fases que es van a seguir són: Introducció, Planificació, Gestió, Anàlisi, Disseny, Implementació, Verificació i Validació i Tancament del projecte. Aquestes fases, tasques i activitats estan descrites en la gestió de l'abast i del temps que s'expliquen més endavant.

### 2.3 Gestió

En aquesta secció s'explica la gestió realitzada d'aquest projecte, en concret es detalla la gestió de l'abast i del temps.

### 2.3.1 Gestió de l'abast

En la gestió de l'abast es detalla els paquets de tasques que es realitzen per al desenvolupament d'aquest projecte, definits en un diagrama EDT (*Estructura de desglòs del treball*) que apareix en la Figura 2.1. Per a cada paquet es descriuen les activitats a realitzar i els responsables d'aquest paquet; tot açò recollit en el diccionari del diagrama EDT que es mostra a les taules compreses entre la Taula 3.1 i la Taula 2.9.

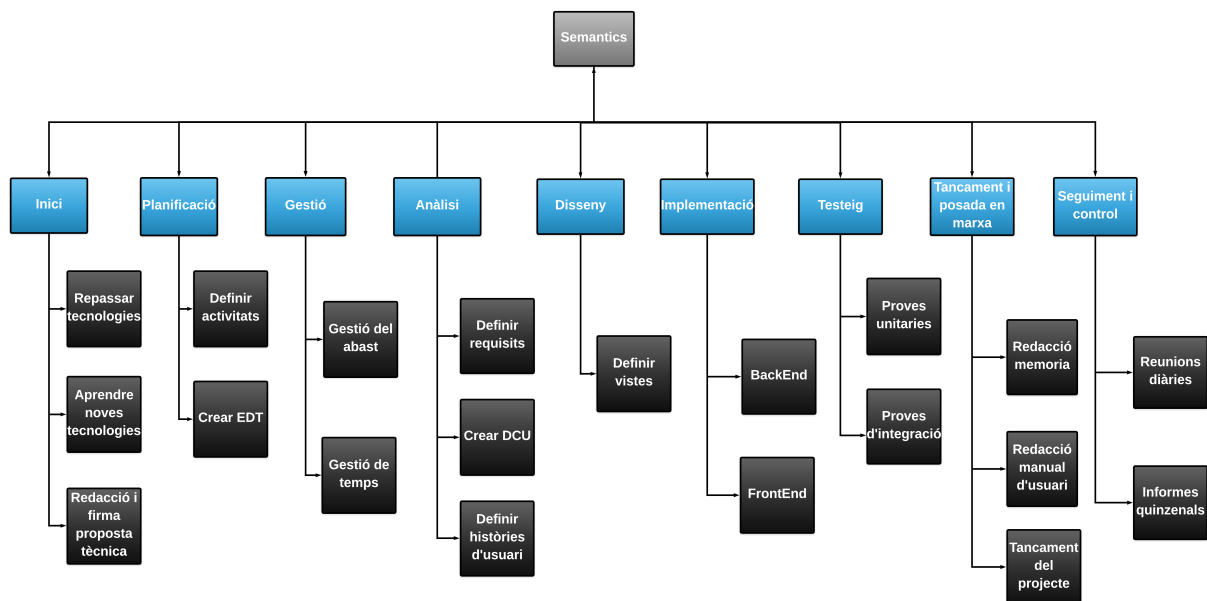


Figura 2.1: Diagrama EDT.

Codi del paquet de treball	Nom del paquet de treball
1.	Inici
<b>Objectiu del paquet:</b>	Iniciar el projecte
<b>Descripció del paquet de treball:</b>	Primer paquet de desenvolupament del projecte. Es realitzaran les tasques per a iniciar el projecte.
<b>Descripció del treball a realitzar (activitats):</b>	<ul style="list-style-type: none"> <li>• Repassar Tecnologies</li> <li>• Aprendre noves tecnologies</li> <li>• Redacció i firma de la proposta tècnica</li> </ul>
<b>Assignació de responsabilitats:</b>	<b>Responsable:</b> Mario <b>Aprova:</b> Supervisor i Tutor <b>Consultat:</b> Supervisor <b>Informat:</b> Supervisor
<b>Dates programades:</b>	<b>Inici:</b> 21/02/2020 <b>Fi:</b> 06/03/2020
<b>Criteris d'acceptació:</b>	<b>Responsable d'acceptació:</b> Tutor <b>Requisits que s'han de complir:</b> La proposta tècnica ha d'incloure la descripció, objectius, tasques i la planificació inicial del projecte. <b>Forma en que s'acceptarà:</b> Corregint la proposta tècnica i modificant el estat de la tasca a "Satisfactori"
<b>Riscos:</b>	Que la proposta tècnica no siga acceptada.
<b>Mesures de contingència:</b>	<ul style="list-style-type: none"> <li>• Realitzar la planificació inicial de forma correcta</li> <li>• Realitzar una serie d'activitats de seguiment i control</li> </ul>
<b>Mesures pal·liatives:</b>	<ul style="list-style-type: none"> <li>• Remarcar el canvi de la planificació inicial dins d'un informe quinzenal</li> <li>• Tornar a realitzar la proposta tècnica</li> </ul>

Taula 2.1: Diccionari EDT (Introducció)

Codi del paquet de treball	Nom del paquet de treball
2.	Planificació
<b>Objectiu del paquet:</b>	Planificar el projecte
<b>Descripció del paquet de treball:</b>	En aquest paquet es realitzen les tasques per a dur a terme la planificació inicial del projecte.
<b>Descripció del treball a realitzar (activitats):</b>	<ul style="list-style-type: none"> <li>• Definir activitats</li> <li>• Crear EDT</li> </ul>
<b>Assignació de responsabilitats:</b>	<b>Responsable:</b> Mario <b>Aprova:</b> Supervisor i Tutor <b>Consultat:</b> Supervisor <b>Informat:</b> Supervisor
<b>Dates programades:</b>	<b>Inici:</b> 24/02/2020 <b>Fi:</b> 03/03/2020
<b>Criteris d'acceptació:</b>	<b>Responsable d'acceptació:</b> Tutor <b>Requisits que s'han de complir:</b> Les activitats han d'estar ben definides d'un principi al igual que els paquets del diagrama EDT <b>Forma en que s'acceptarà:</b> Aprovant la proposta tècnica
<b>Riscos:</b>	Que es produísca una mala planificació
<b>Mesures de contingència:</b>	<ul style="list-style-type: none"> <li>• Seguir la planificació inicial de la proposta tècnica</li> <li>• Realitzar una serie d'activitats de seguiment i control</li> </ul>
<b>Mesures pal·liatives:</b>	<ul style="list-style-type: none"> <li>• Remarcar el canvi de la planificació inicial dins d'un informe quinzenal</li> <li>• Editar la planificació i notificar aquests canvis al tutor</li> </ul>

Taula 2.2: Diccionari EDT (Planificació)

Codi del paquet de treball	Nom del paquet de treball
<b>3.</b>	Gestió
<b>Objectiu del paquet:</b>	Gestionar la realització del projecte
<b>Descripció del paquet de treball:</b>	Es realitzaran les tasques per a gestionar el desenvolupament del projecte. En concret l'abast i la gestió del temps.
<b>Descripció del treball a realitzar (activitats):</b>	<ul style="list-style-type: none"> <li>• Gestió de l'abast</li> <li>• Gestió del temps</li> </ul>
<b>Assignació de responsabilitats:</b>	<b>Responsable:</b> Mario <b>Aprova:</b> Supervisor i Tutor <b>Consultat:</b> Supervisor <b>Informat:</b> Supervisor i Tutor
<b>Dates programades:</b>	<b>Inici:</b> 26/02/2020 <b>Fi:</b> 04/03/2020
<b>Criteris d'acceptació:</b>	<b>Responsable d'acceptació:</b> Tutor <b>Requisits que s'han de complir:</b> El diagrama <i>Gantt</i> ha de incloure les activitats i paquets que es mostren en la gestió del abast. <b>Forma en que s'acceptarà:</b> Aprovant la proposta tècnica
<b>Riscos:</b>	<ul style="list-style-type: none"> <li>• Que la gestió de l'abast no estiga ben definida</li> <li>• Que la gestió del temps siga massa optimista</li> </ul>
<b>Mesures de contingència:</b>	<ul style="list-style-type: none"> <li>• Realitzar una sèrie de reunions amb el supervisor</li> <li>• Preguntar al supervisor si sorgeix qualsevol dubte sobre el projecte</li> </ul>
<b>Mesures pal·liatives:</b>	<ul style="list-style-type: none"> <li>• Editar l'abast del projecte i notificar al supervisor i al tutor</li> <li>• Editar la gestió del temps i notificar el canvi en els informes quinzenals</li> </ul>

Taula 2.3: Diccionari EDT (Gestió)

Codi del paquet de treball	Nom del paquet de treball
4.	Anàlisi
<b>Objectiu del paquet:</b>	Analitzar el sistema a realitzar
<b>Descripció del paquet de treball:</b>	Es realitzarà l'anàlisi del sistema a desenvolupar.
<b>Descripció del treball a realitzar (activitats):</b>	<ul style="list-style-type: none"> <li>• Definició de requisits</li> <li>• Diagrama de casos d'ús</li> <li>• Històries d'usuari</li> </ul>
<b>Assignació de responsabilitats:</b>	<b>Responsable:</b> Mario <b>Aprova:</b> Supervisor i Tutor <b>Consultat:</b> Supervisor <b>Informat:</b> Supervisor
<b>Dates programades:</b>	<b>Inici:</b> 04/03/2020 <b>Fi:</b> 05/03/2020
<b>Criteris d'acceptació:</b>	<b>Responsable d'acceptació:</b> Supervisor <b>Requisits que s'han de complir:</b> El anàlisi del sistema ha de reflectir el sistema real <b>Forma en que s'acceptarà:</b> El supervisor donarà l'aprovació i es passarà a la fase següent
<b>Riscos:</b>	<ul style="list-style-type: none"> <li>• Que no s'accepte la definició dels requisits del sistema</li> <li>• Que no s'accepte ni el diagrama de casos d'ús ni la seua especificació</li> </ul>
<b>Mesures de contingència:</b>	<ul style="list-style-type: none"> <li>• Realitzar una sèrie de reunions amb el supervisor</li> <li>• Preguntar al supervisor si sorgeix qualsevol dubte sobre el projecte</li> </ul>
<b>Mesures pal·liatives:</b>	<ul style="list-style-type: none"> <li>• Editar els requisits i el diagrama segons les indicacions del supervisor</li> <li>• Notificar aquest canvi en els informes quinzenals</li> </ul>

Taula 2.4: Diccionari EDT (Anàlisi)



Codi del paquet de treball	Nom del paquet de treball
5.	Disseny
<b>Objectiu del paquet:</b>	Dissenyar la interfície gràfica del projecte
<b>Descripció del paquet de treball:</b>	Es realitzaran prototipus de la interfície gràfica d'usuari. Aquesta interfície d'usuari serà part del projecte però no del producte, perquè s'utilitzarà per a realitzar la demostració en la presentació final del tfg.
<b>Descripció del treball a realitzar (activitats):</b>	<ul style="list-style-type: none"> <li>• Disseny d'interfícies d'usuari</li> </ul>
<b>Assignació de responsabilitats:</b>	<b>Responsable:</b> Mario <b>Aprova:</b> Supervisor <b>Consultat:</b> Supervisor <b>Informat:</b> Supervisor
<b>Dates programades:</b>	<b>Inici:</b> 05/03/2020 <b>Fi:</b> 10/03/2020
<b>Criteris d'acceptació:</b>	<b>Responsable d'acceptació:</b> Supervisor <b>Requisits que s'han de complir:</b> Ha de seguir un disseny coherent entre les vistes <b>Forma en que s'acceptarà:</b> El supervisor donarà l'aprovació i es podrà tancar la tasca correctament
<b>Riscos:</b>	Que el disseny realitzat no siga agradable per al usuari
<b>Mesures de contingència:</b>	<ul style="list-style-type: none"> <li>• Realitzar una sèrie de reunions amb el supervisor i l'equip de l'empresa</li> <li>• Mostrar els prototipus al supervisor</li> </ul>
<b>Mesures pal·liatives:</b>	<ul style="list-style-type: none"> <li>• Editar els prototipus</li> <li>• Notificar aquest canvi al supervisor</li> </ul>

Taula 2.5: Diccionari EDT (Disseny)

Codi del paquet de treball	Nom del paquet de treball
6.	Implementació
<b>Objectiu del paquet:</b>	Implementar el sistema ha realitzar del projecte
<b>Descripció del paquet de treball:</b>	Es realitzarà la implementació dels diferents mòduls del projecte.
<b>Descripció del treball a realitzar (activitats):</b>	<ul style="list-style-type: none"> <li>● Back-End <ul style="list-style-type: none"> <li>○ Mòdul Estudis</li> <li>○ Mòdul Informes</li> <li>○ Mòdul Neteja</li> <li>○ Mòdul Càrrega</li> </ul> </li> <li>● Front-End <ul style="list-style-type: none"> <li>○ Vista <i>login</i></li> <li>○ Vista buscador</li> </ul> </li> </ul>
<b>Assignació de responsabilitats:</b>	<b>Responsable:</b> Mario <b>Aprova:</b> Supervisor <b>Consultat:</b> Supervisor <b>Informat:</b> Supervisor
<b>Dates programades:</b>	<b>Inici:</b> 06/03/2020 <b>Fi:</b> 08/05/2020
<b>Criteris d'acceptació:</b>	<b>Responsable d'acceptació:</b> Supervisor <b>Requisits que s'han de complir:</b> El programa ha d'estar escrit en llenguatge <i>Python</i> i ha de poder netejar els informes de qualsevol client. <b>Forma en que s'acceptarà:</b> El supervisor donarà l'aprovació i es podrà tancar la tasca correctament
<b>Riscos:</b>	Que el cost computacional del codi siga alt
<b>Mesures de contingència:</b>	<ul style="list-style-type: none"> <li>● Realitzar una sèrie de reunions amb el supervisor i l'equip de l'empresa</li> <li>● Realitzar una programació neta</li> </ul>
<b>Mesures pal·liatives:</b>	<ul style="list-style-type: none"> <li>● Refactoritzar el codi</li> <li>● Notificar aquest canvi al supervisor</li> </ul>

Taula 2.6: Diccionari EDT (Implementació)

Codi del paquet de treball	Nom del paquet de treball
7.	Testeig
<b>Objectiu del paquet:</b>	Realitzar les proves del sistema implementat
<b>Descripció del paquet de treball:</b>	Es realitzaran els codis de prova per a testar el funcionament de la fase d'implementació
<b>Descripció del treball a realitzar (activitats):</b>	<ul style="list-style-type: none"> <li>• Proves unitàries</li> <li>• Proves d'integració</li> </ul>
<b>Assignació de responsabilitats:</b>	<b>Responsable:</b> Mario <b>Aprova:</b> Supervisor i Tutor <b>Consultat:</b> Supervisor <b>Informat:</b> Supervisor
<b>Dates programades:</b>	<b>Inici:</b> 19/05/2020 <b>Fi:</b> 27/05/2020
<b>Criteris d'acceptació:</b>	<b>Responsable d'acceptació:</b> Supervisor <b>Requisits que s'han de complir:</b> Han de passar totes les proves implementades així com hi haver proves per a provar la connexió amb el buscador semàntic i proves per a comprovar els resultats tornats per aquest. <b>Forma en que s'acceptarà:</b> S'acceptarà quan es superen tots els test de manera satisfactòria
<b>Riscos:</b>	Que el codi implementat en la fase anterior no passe les proves
<b>Mesures de contingència:</b>	<ul style="list-style-type: none"> <li>• Realitzar una sèrie de reunions amb el supervisor</li> <li>• Realitzar tests que proven tota la funcionalitat</li> </ul>
<b>Mesures pal·liatives:</b>	<ul style="list-style-type: none"> <li>• Refactoritzar el codi</li> <li>• Refactoritzar els tests</li> <li>• Notificar aquest canvi al supervisor</li> </ul>

Taula 2.7: Diccionari EDT (Testeig).

Codi del paquet de treball	Nom del paquet de treball
8.	Tancament i posada en marxa
<b>Objectiu del paquet:</b>	Tancar el projecte i posar en marxa el producte resultat d'aquest
<b>Descripció del paquet de treball:</b>	Les activitats a realitzar seran les últimes del desenvolupament del projecte.
<b>Descripció del treball a realitzar (activitats):</b>	<ul style="list-style-type: none"> <li>• Tancar el projecte</li> <li>• Redacció memòria</li> <li>• Redacció manual d'usuari</li> </ul>
<b>Assignació de responsabilitats:</b>	<b>Responsable:</b> Mario <b>Aprova:</b> Supervisor i Tutor <b>Consultat:</b> Supervisor <b>Informat:</b> Supervisor
<b>Dates programades:</b>	<b>Inici:</b> 29/05/2020 <b>Fi:</b> 29/05/2020
<b>Criteris d'acceptació:</b>	<b>Responsable d'acceptació:</b> Supervisor <b>Requisits que s'han de complir:</b> El manual ha de ser clar i ajudar a la instal·lació dels scripts i al seu executament. <b>Forma en que s'acceptarà:</b> El supervisor seguirà el manual i provarà els scripts implementats en la fase d'implementació.
<b>Riscos:</b>	<ul style="list-style-type: none"> <li>• Que el producte final no siga acceptat pels clients</li> <li>• Que no s'accepte la memòria final</li> </ul>
<b>Mesures de contingència:</b>	<ul style="list-style-type: none"> <li>• Realitzar una sèrie de reunions amb el supervisor i l'equip de l'empresa</li> <li>• Entregar esborranys de la memòria al tutor</li> </ul>
<b>Mesures pal·liatives:</b>	<ul style="list-style-type: none"> <li>• Tancar el projecte però no posar-lo en marxa</li> <li>• Realitzar les correccions de la memòria suggerides per el tutor</li> </ul>

Taula 2.8: Diccionari EDT (Tancament i posada en marxa).

Codi del paquet de treball	Nom del paquet de treball
9.	Seguiment i control
<b>Objectiu del paquet:</b>	Realitzar activitats de seguiment i control del desenvolupament del projecte
<b>Descripció del paquet de treball:</b>	
<b>Descripció del treball a realitzar (activitats):</b>	<ul style="list-style-type: none"> <li>• Informes</li> <li>• Reunions</li> </ul>
<b>Assignació de responsabilitats:</b>	<b>Responsable:</b> Mario i Supervisor <b>Aprova:</b> Supervisor i Tutor <b>Consultat:</b> Supervisor <b>Informat:</b> Supervisor i Tutor
<b>Dates programades:</b>	<b>Inici:</b> 21/02/2020 <b>Fi:</b> 29/05/2020
<b>Criteris d'acceptació:</b>	<b>Responsable d'acceptació:</b> Tutor <b>Requisits que s'han de complir:</b> Els informes s'han d'entregar cada 15 dies i han de contenir informació sobre la quinzena anterior, l'actual i la pròxima <b>Forma en que s'acceptarà:</b> El tutor corregirà i donarà retroacció als informes.
<b>Riscos:</b>	Que no s'accepten els informes quinzenals
<b>Mesures de contingència:</b>	<ul style="list-style-type: none"> <li>• Realitzar una sèrie de reunions amb el supervisor i l'equip de l'empresa</li> <li>• Realitzar informes que mostren les fases desenvolupades i les que falten per desenvolupar</li> </ul>
<b>Mesures pal·liatives:</b>	<ul style="list-style-type: none"> <li>• Modificar la forma de redactar els informes</li> </ul>

Taula 2.9: Diccionari EDT (Seguiment i control).

### 2.3.2 Gestió del temps

En aquesta fase s'expliquen les activitats a realitzar que componen els paquets descrits anteriorment en la gestió de l'abast. Es detalla també, la planificació d'aquestes activitats mitjançant un diagrama de *Gantt* (Figura 2.2). El diagrama de *Gantt* és una de les tècniques utilitzades tant per l'administració pública com per l'empresa privada com a eina de planificació del treball.

Les activitats portades a terme en aquest projecte són les següents:

1. Inici
  - (a) Repassar tecnologies
    - i. Git
    - ii. Python
  - (b) Aprendre noves tecnologies
    - i. MySQL
  - (c) Redacció proposta tècnica
2. Planificació
  - (a) Definir activitats
  - (b) Crear diagrama EDT
3. Gestió
  - (a) Gestió del temps
  - (b) Gestió de l'abast
4. Anàlisi
  - (a) Definir requisits
  - (b) Crear DCU
5. Disseny
  - (a) Disseny vista *Login*
  - (b) Disseny vista Buscador
6. Implementació
  - (a) *BackEnd*
    - i. Mòdul Estudis
    - ii. Mòdul Informes
    - iii. Mòdul Neteja
    - iv. Mòdul Càrrega
  - (a) *FrontEnd*
    - i. Vista *Login*

ii. Vista Buscador

7. Testeig

- (a) Proves d'unitat
- (b) Proves d'integració

8. Tancament i posada en marxa

- (a) Redactar manual d'usuari
- (b) Tancar el projecte

9. Seguiment i control

- (a) Reunions diàries
- (b) Informes quinzenals

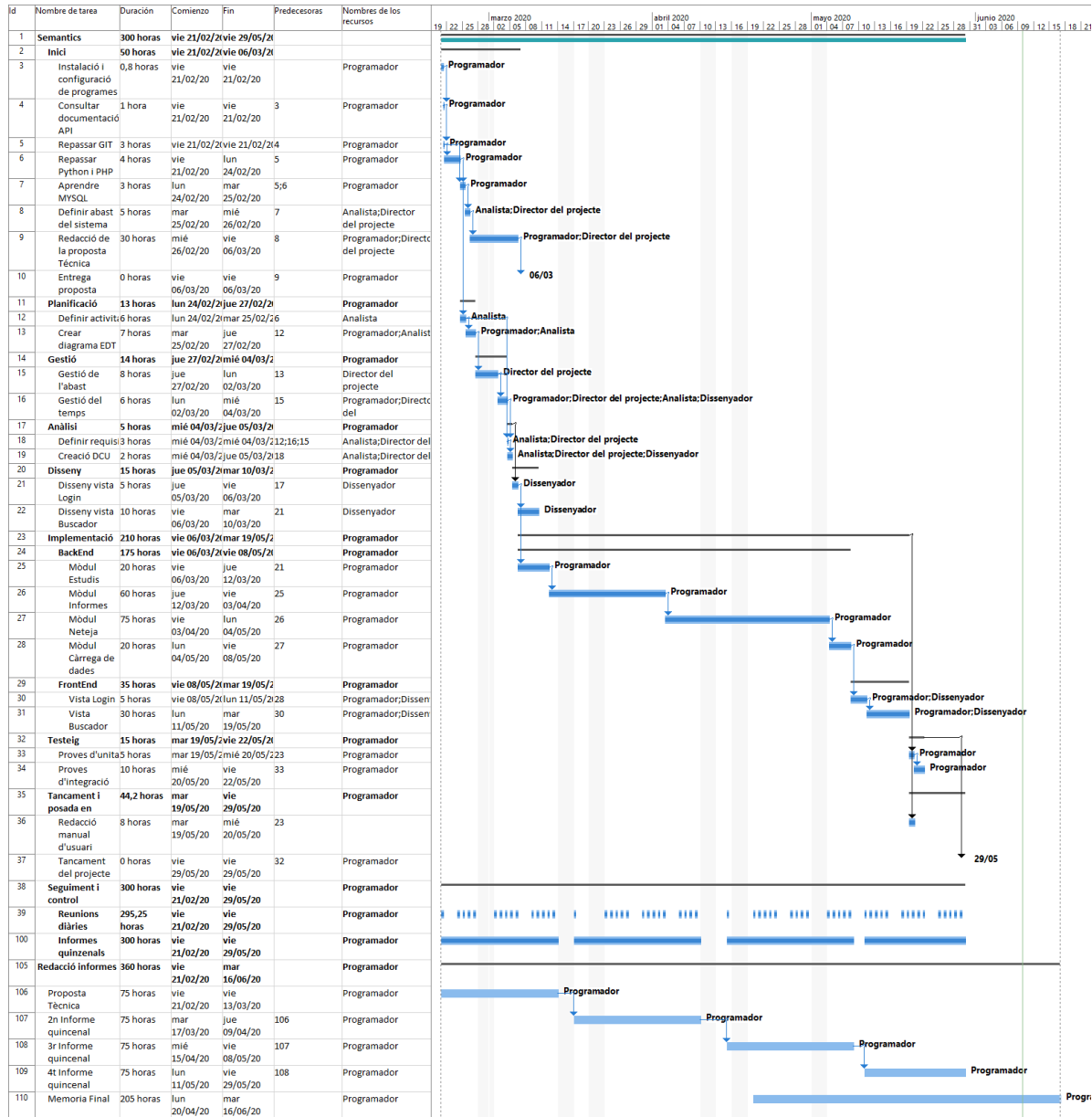


Figura 2.2: Diagrama Gantt.



## 2.4 Seguiment del projecte

Les activitats realitzades dins d'aquesta fase ajuden al seguiment i control del desenvolupament del projecte. Aquestes activitats són: reunions diàries i informes quinzenals.

La durada de les reunions comprén al voltant de 15 minuts i es realitzen tots el dies. En aquestes reunions participen tots els membres de l'empresa i expliquen el que han fet, el que estan fent, el que faran i els problemes que han tingut al llarg del dia de treball anterior. Aquestes reunions s'utilitzen per al control i seguiment del projecte ja que tots els programadors de l'empresa ajuden a resoldre els problemes que han aparegut.

Els informes es realitzen cada 15 dies i en ells s'explica: el treball realitzat en la quinzena anterior, el treball realitzat en la quinzena actual i el treball que es pretén realitzar en la quinzena següent. Aquests informes també contenen un camp per anotar observacions durant la realització de les pràctiques. En concret s'ha detallat el treball realitzat, com s'anaven implementant els mòduls i algunes solucions o millores que van facilitar la realització d'aquest projecte.

### **Quinzena actual del tercer informe quinzenal:**

*"En aquesta quinzena vaig acabar alguns dels programes del Backend de la fase d'Implementació:*

1. *Mòdul Estudis.*
2. *Mòdul Informes.*
3. *Mòdul Neteja.*

*Com a resultat del primer mòdul, es va crear un codi capaç de recollir tots els `study_iuid` d'un client i es van classificar en fitxers per any de creació de l'estudi.*

*Després vaig implementar el segon mòdul, què s'encarrega d'agafar aquestos `study_iuid` i fer una cridada a la base de dades per agafar la modalitat i la descripció d'aquest estudi. També amb aquest `id`, fer una cridada a una api per agafar el informes d'aquest estudi (si ne té). I amb tot açò crea una línia en un fitxer per a cada estudi amb informe de la següent manera: `id desc mod informe`*

*Per últim del mòdul neteja, es va crear un codi que agafa els informes i els neteja d'informació sensible com poden ser noms propis, llocs de residència, etc. Aquest últim mòdul és el que em queda per acabar i això ho faré en la pròxima quinzena."*

Un d'aquests informes quinzenals, en concret el primer informe quinzenal, conté la proposta tècnica d'aquest projecte. En aquesta es detalla la descripció i la planificació inicial del projecte. Durant el desenvolupament del projecte no han hi hagut grans canvis, no obstant, es van suprimir o canviar tasques com "Proves d'interfície d'usuari" per "Redactar manual d'usuari", i s'han afegit algunes tecnologies més que no es van tindre en compte com per exemple Django i Javascript.

Si en algun moment s'ha canviat la planificació, s'explica en els informes quinzenals, descrivint els canvis i la nova planificació així com el diagrama de *Gantt*. A partir del tercer informe quinzenal, els següents informes es van realitzar des de casa, a causa de la situació de la COVID-19.

Una vegada acabats tots els informes quinzenals, s'ha realitzat la redacció d'aquesta memòria final.

## Capítol 3

# Anàlisi i disseny del sistema

### 3.1 Anàlisi del sistema

El sistema a realitzar constarà de una sèrie de programes en Python que tindran com a finalitat recollir els estudis i els informes de qualsevol client de l'empresa; i, d'aquests informes netejar qualsevol informació sensible. Aquest sistema està compostat per 3 programes que realitzaran aquestes funcionalitats i unes 4 classes per a comprovar la connexió amb l'API del buscador.

Els informes i estudis dels clients ja es troben en una base de dades, per tant, un dels programes realitzats té contacte amb aquesta base de dades per a poder recollir els estudis. Per a recollir els informes, el mòdul Informes realitza cridades a l'API d'ActualPacs del client amb *l'study\_iuid* de l'estudi per recollir l'informe.

Un altre dels mòduls, en concret el mòdul Càrrega manté contacte amb l'API del buscador semàntic per a poder recollir i carregar informes nous a l'API.

Aquests programes formen el sistema a codificar en aquest projecte. El més important de tot és que siguin capaços de funcionar per a qualsevol client de l'empresa. Han de ser suficientment genèrics i modificables per a poder netejar els informes de tots els clients sense modificar el codi, només un fitxer de configuració del sistema.

Per a comprovar el funcionament del codi, es realitzarà un sistema de *log* on s'aniran registrant tots els informes i passos que van fent els programes així com errors que poden sorgir. Aquests fitxers *logs* es creen cada 24 hores per a que no s'escriuen en un sol fitxer.

#### 3.1.1 Requisits

Els usuaris del buscador semàntic són els usuaris d'ActualPacsv4, en concret, radiòlegs. Aquests podran realitzar cerques d'articles i/o informes passant una o unes paraules. L'objectiu d'aquest

programa serà alimentar <sup>1</sup> aquest buscador d'informes nets de dades sensibles. Aquestes dades poden ser el nom del pacient tractat, tant com, el doctor o metge que l'ha tractat. A més, aquestes dades sensibles poden incloure números de telèfon, direccions de correu, etc.

El programa a implementar ha de ser capaç de:

- Recollir els estudis i classificar-los.
- Recollir els informes d'aquests estudis.
- Netejar els informes de dades sensibles.
- Carregar els informes nets a l'aplicació.

### 3.1.2 Diagrama de casos d'ús

En aquest diagrama (Figura 3.1) es mostren els actors del sistema del buscador semàntic amb la implementació d'aquest projecte. Els actors son els següents:

- Usuari: són els radiòlegs clients de l'empresa. Inicia sessió en la plataforma i realitza una cerca d'informes o d'articles.
- SemanticScript: és el codi realitzat en aquest projecte. Neteja els informes i els carrega a l'API del buscador semàntic.

---

<sup>1</sup>Argot de l'empresa: significa carregar el buscador.

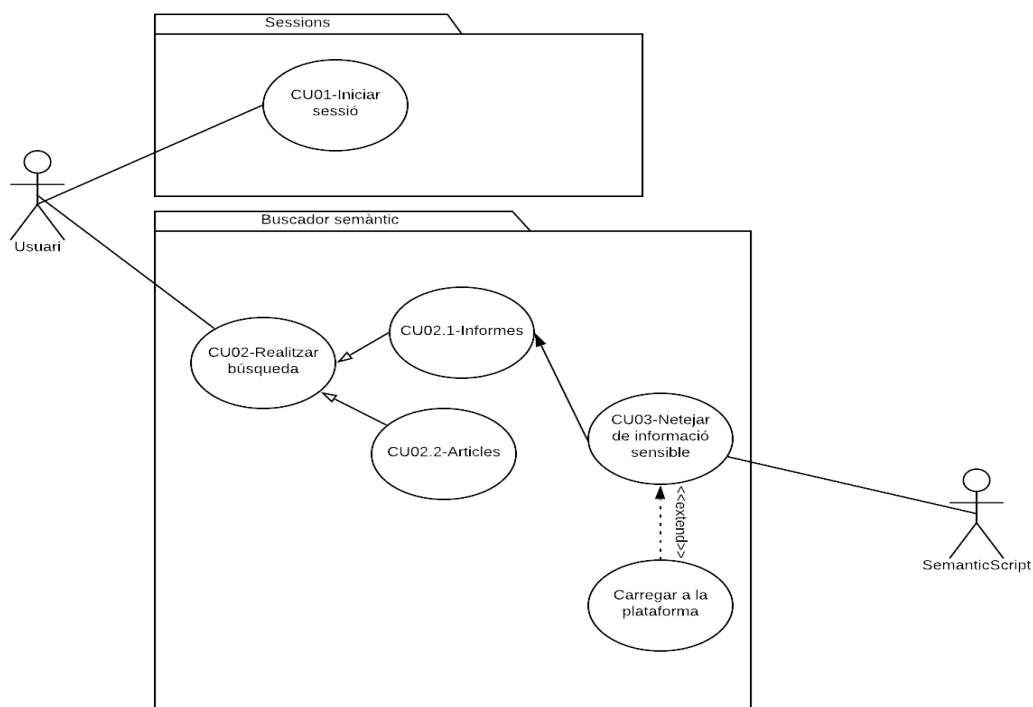


Figura 3.1: Diagrama de casos d'ús.

Especificació del cas d'ús	
<b>Identificador</b>	CU01
<b>Nom</b>	Iniciar sessió
<b>Versió</b>	1.0
<b>Descripció</b>	El sistema ha de permetre als usuaris registrats iniciar sessió en la plataforma
<b>Abast</b>	Fins que el client introdueix el seu correu electrònic i la seva contrasenya i entre a la web del buscador semàntic
<b>Nivell</b>	Subtasca
<b>Actor principal</b>	Usuari
<b>Actors secundaris</b>	No n'hi ha
<b>Relacions</b>	No n'hi ha
<b>Precondició</b>	L'usuari ha d'estar registrat en la base de dades de la plataforma

*continued on next page*

*continued from previous page*

<b>Condicció final amb èxit</b>	L'usuari iniciarà sessió i entrarà al buscador semàntic
<b>Condicció final amb fracàs</b>	L'usuari no està registrat i no podrà iniciar sessió
<b>Trigger</b>	Poder accedir a la web del buscador semàntic
<b>Seqüència normal</b>	<b>Acció</b>
<b>1</b>	L'usuari entra a la web del <i>login</i>
<b>2</b>	L'usuari introdueix el seu correu electrònic i la seva contrasenya
<b>3</b>	Les dades de l'usuari són correctes i entra a la web del buscador semàntic
<b>Excepcions &lt;acció 3&gt;</b>	<b>Excepció dades incorrectes</b>
<b>1</b>	Les dades de l'usuari són incorrectes
<b>2</b>	L'usuari no està registrat en la plataforma
<b>Freqüència esperada</b>	Diària
<b>Importància</b>	Necessari
<b>Prioritat</b>	Curt termini

Taula 3.1: Especificació cas d'ús (CU01)

Especificació del cas d'ús	
<b>Identificador</b>	CU02
<b>Nom</b>	Realitzar búsqueda
<b>Versió</b>	1.0
<b>Descripció</b>	El buscador semàntic deu permetre realitzar una cerca d'informes o articles al introduir una o més paraules
<b>Abast</b>	Fins que l'usuari introdueix el terme de búsqueda i el buscador mostra els resultats que ha obtingut cridant a l'API del buscador amb eixe terme com a paràmetre

*continued on next page*

*continued from previous page*

<b>Nivell</b>	Tasca principal
<b>Actor principal</b>	Usuari
<b>Actors secundaris</b>	No n'hi ha
<b>Relacions</b>	CU02.1-Informes, CU02.2-Articles
<b>Precondició</b>	L'usuari té iniciar sessió per a poder accedir al buscador semàntic
<b>Condicció final amb èxit</b>	El sistema mostrarà una taula amb la informació dels informes/articles amb la paraula que ha introduït l'usuari
<b>Condicció final amb fracàs</b>	El sistema no mostrarà cap informació perquè la paraula no és correcta o no es troba cap informe/article amb aquesta
<b>Trigger</b>	Poder cercar informes o articles radiològics que siguin d'ajuda per a l'usuari
<b>Seqüència normal</b>	<b>Acció</b>
<b>1</b>	L'usuari introdueix una o més paraules
<b>2</b>	L'usuari elegix si vol cercar informes o articles
<b>3</b>	L'usuari li dona a cercar
<b>4</b>	El sistema mostra un grafisme indicant que està realitzant la cerca
<b>5</b>	Finalment el sistema mostra una taula amb els informes/articles
<b>Excepcions &lt;acció 1&gt;</b>	<b>Excepció paraula incorrecta</b>
<b>1</b>	Si l'usuari fica un caràcter incorrecte
<b>2</b>	Si la paraula que ha introduït l'usuari no es correspon a cap informe
<b>Freqüència esperada</b>	Diària
<b>Importància</b>	Necessari
<b>Prioritat</b>	Curt termini

Taula 3.2: Especificació cas d'ús (CU02)

Especificació del cas d'ús	
<b>Identificador</b>	CU03
<b>Nom</b>	Netejar d'informació sensible
<b>Versió</b>	1.0
<b>Descripció</b>	El sistema SemanticScript ha de poder netejar informes radiològics de dades personals dels doctors i pacients per a la posterior càrrega en el buscador semàntic
<b>Abast</b>	Fins recollir els estudis, i d'aquests els informes, a netejar aquests informes i carregar-los en la API del buscador semàntic
<b>Nivell</b>	Tasca principal
<b>Actor principal</b>	SemanticScript
<b>Actors secundaris</b>	No n'hi ha
<b>Relacions</b>	CU02.1-Informes, «extend» Carregar a la plataforma
<b>Precondició</b>	Han d'existir informes que netejar i carregar a la API
<b>Condicció final amb èxit</b>	El sistema mostrarà una taula amb la informació dels informes lliure de dades personals de doctors i pacients
<b>Condicció final amb fracàs</b>	El sistema mostrarà una taula amb la informació dels informes amb dades personals de doctors i pacients
<b>Trigger</b>	Poder carregar al buscador informes radiològics lliures d'informació sensible dels doctors i pacients
<b>Seqüència normal</b>	<b>Acció</b>
<b>1</b>	SemanticScript recull tots els estudis radiològics
<b>2</b>	SemanticScript recull tots els informes d'aquests estudis
<b>3</b>	SemanticScript neteja aquests informes de dades personals

*continued on next page*



*continued from previous page*

<b>4</b>	SemanticScript carrega els informes nets a l'API del buscador
<b>Excepcions &lt;acció 3&gt;</b>	<b>Excepció informe brut</b>
<b>1</b>	Si SemanticScript no neteja bé alguns informes
<b>Freqüència esperada</b>	Annual
<b>Importància</b>	Necessari
<b>Prioritat</b>	Mig termini

Taula 3.3: Especificació cas d'ús (CU03)

### 3.1.3 Històries d'usuari

*As a radiòleg*

*When* introduïsc “rotura de fèmur”

*Then* m'apareix un llistat de informes i articles sobre estudis de trencament de fèmur.

*As a radiòleg*

*When* trie que només isquen informes i introduïsc “Pierna izquierda”

*Then* m'apareix un llistat de informes sobre estudis on apareix aquesta paraula.

*As a radiòleg*

*When* trie que només isquen articles i introduïsc “Tórax”

*Then* m'apareix un llistat de articles sobre estudis on apareix aquesta paraula.

*As a clínica de radiologia*

*When* carrego els informes a la base de dades

*Then* el buscador mostra els informes nets de informació sensible.

## 3.2 Disseny de l'arquitectura del sistema

El disseny del sistema així com el disseny de la base de dades, ha sigut dissenyat prèviament per l'equip de l'empresa. En aquest projecte vaig utilitzar les bases de dades dels clients, en un sistema local, per a poder realitzar el codi i les proves.

### 3.3 Disseny de la interfície

En aquesta fase es van realitzar els prototipus de les vistes per a la demostració tècnica d'aquest projecte. Es van realitzar els prototipus de la vista del *login* i del buscador semàntic.

Per realitzar aquests prototipus es va utilitzar la web Figma [11]. Es van dissenyar dos prototipus seguint un poc un exemple que em va donar des de l'empresa. Aquest exemple constava del buscador oficial i la pàgina de *login* per accedir a ell.

L'estil elegit va ser elecció meua. Es van utilitzar colors grisos i blaus per a donar-li un estil més sobri, i per a saber quins colors utilitzar dins del gradient de blaus i grisos es va utilitzar la web Adobe Color [12]. Aquesta web proporciona una roda de colors i es poden buscar colors complementaris al color que es vulga elegir.

La primera d'aquestes vistes és la del *login* (Figura 3.2), que conté una imatge i un formulari per a iniciar sessió en la plataforma.

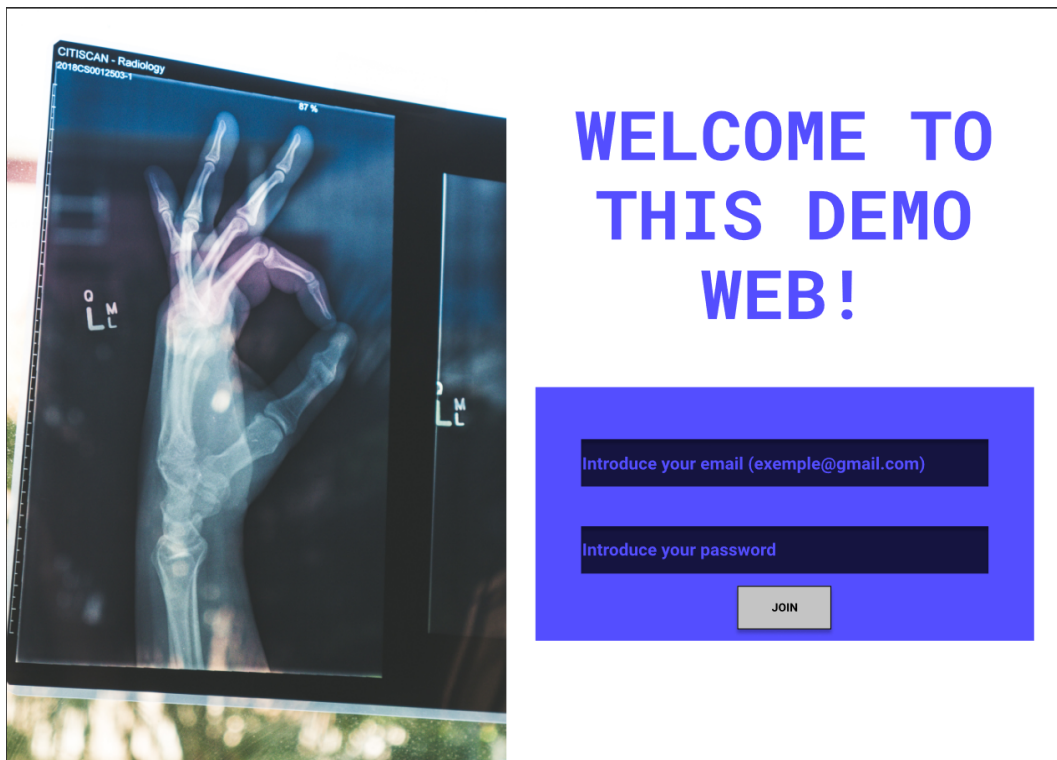


Figura 3.2: Vista Login.

La segona vista és la del buscador (Figura 3.3), que conté una barra de cerca, un botó per a iniciar aquesta cerca i una taula amb els camps dels informes. Aquesta vista ha sigut la que més canviï durant la realització d'aquest projecte, però va ajudar a tindre una idea general de com deuria ser.



The image shows a user interface for a search function. At the top, there is a dark blue search bar with the placeholder text "Introduce un término de búsqueda" and a light gray search button to its right. Below the search bar is a table with a dark blue header and a light blue body. The table has four columns: "id", "modalidad", "descripción", and "informe". The table is currently empty of data rows.

id	modalidad	descripción	informe
----	-----------	-------------	---------

Figura 3.3: Vista Buscador.

Aquests dissenys d'interfícies i implementació posterior no entren dins de l'abast del producte ja que ja està implementat en l'empresa. Però sí que entren dins de l'abast del projecte, ja que es va a realitzar una demostració per a poder exposar millor aquest projecte.



## Capítol 4

# Implementació i proves

### 4.1 Detalls d'implementació

En aquesta fase es va implementar el codi necessari per a poder realitzar aquest projecte. En total es van escriure 4 mòduls que són els que realitzaran la funcionalitat de recollir, classificar estudis i informes, netejar informes i pujar aquestos al buscador semàntic. En tots aquest projecte s'utilitza un fitxer auxiliar de configuració (*config\_default.py*), d'aquesta manera, el programa és modificable des de fóra sense editar ni eliminar codi dels mòduls. Amb aquest fitxer de configuració s'ha pogut realitzar els següents programes, funcionals per a tots els clients d'ActualTec.

#### 4.1.1 Mòdul Estudis

El fitxer *classify\_study.py* recull els identificadors dels estudis radiològics per any i els classifica en fitxers amb l'any del estudi.

*classify\_study.py*: Al executar el programa es demanarà l'any dels estudis que es desitgen recopilar.

- En el *main* es crida a un mètode *dir\_comprobation* que mira si la carpeta amb el nom del client, que hi ha a l'arxiu de configuració com la variable CLIENT, està creada. Si aquesta carpeta està creada, mira dins d'aquesta carpeta. Comprova si ja està creat un arxiu amb el any dels estudis que es van a recollir i, si és així, s'elimina. Es creen també les carpetes per als *logs* i per als informes que s'utilitzaran més endavant.
- *run\_query* (mètode que es troba en el fitxer *database\_connection*) executa la consulta que li passa el *main*, i torna una llista dels identificadors dels estudis, o *study\_iuid*, ordenada pel temps de creació dels estudis de menor a major i els classifica per anys. Després es crida a *create\_file(data, year)*.

- *create\_file* crea el fitxer, (com a nom, l'any dels estudis), i escriu en cada línia un *study\_iuid* fins que estan tots.

### 4.1.2 Mòdul Informes

Després, el *created\_feedfile.py* recull els informes de cadascun dels estudi classificats anteriorment i crea fitxers amb l'any amb el següent format: id | modalitat | descripció | informe. L'identificador està en el fitxer que ha generat el *classify\_study*, la modalitat i la descripció es troba en la base de dades i l'informe es recull cridant a una API. Si l'estudi no té informe no es guarda i si l'informe no té informació rellevant i és un informe de prova, es guarda a part i no es puja al buscador semàntic.

***created\_feedfile.py***: Al executar es demanarà l'any dels informes que es desitgen recopilar a partir del *study\_iuid* en el fitxer dels estudis, i el número de l'últim estudi revisat. Aquest segon paràmetre és per defecte 0.

- *get\_id\_studies* llegeix tots els *study\_iuid* de l'arxiu que té com a nom l'any que s'ha passat com a primer paràmetre. Després, executa els següents mètodes passant-li els *study\_iuid*, i al final crida a *create\_file(study\_iuid, modalitat, descripció, report)* amb aquestes anteriors dades i crea els arxius en la carpeta *reports* amb aquest format: id | mod | desc | report
- *getSql(id)* realitza una cridada al mètode *run\_query* el qual realitza una consulta a la base de dades i recull el camp modalitat i descripció. Aquests camps poden ser *null*, de manera que si ho són, es reemplacen per una cadena buida.
- *get\_report(id)* crida a la API i torna l'informe d'aquest estudi. Si l'informe conté el caràcter '?' (utilitzat en id | mod | desc | report) es reemplaca per una cadena buida. Així no dóna error després si es fa un *split* i s'agafen els camps de l'informe per separat.

### 4.1.3 Mòdul Neteja

Per últim, el *clean\_file.py* neteja aquests informes d'informació sensible com poden ser noms propis i números de telèfon. Per a aquest programa s'ha creat un fitxer amb paraules i frases que seran les que marquen l'inici i el final de la part de l'informe que s'ha de guardar, ja que tota la informació sensible apareix al principi i al final d'aquest.

Per a la realització d'aquest mòdul es va utilitzar un algorisme d'ús comú en el desenvolupament del software: "Divide y vencerás". No es va seguir l'estructura d'aquest algorisme però en essència és lo que realitza aquest programa. Divideix el informe en parts més senzilles, en aquest cas per tal de trobar un patró.

***clean\_file.py***: Aquest programa et demana l'any de l'informe que es desitja netejar. Després agafant frases i paraules del fitxer *words\_file\_clean.txt* va netejant els informes de l'any que se li ha passat.

Primer, des de el mètode *clean(report)* crida a *clean\_desc(report)*, passant-li l'informe. Aquest mètode mira si en el tercer camp del informe ( id | mod | desc | report), el camp *desc* conté la paraula "DR" o "DRA". Si és així, retalla aquest camp fins eixa paraula i, com a resultat, el camp *desc* acaba net d'aquestes paraules i del nom del doctor o doctora que vindria a continuació d'aquestes.

Després crida a *get\_phrases(file)*, passant-li l'informe, que llegeix i agafa línies del fitxer *words\_file\_clean.txt*:

- PI (paraules inicials)
- PF (paraules finals)
- PS (paraules estranyes)
- PE (paraules a eliminar)

Primer mira si l'informe és de prova o està incomplet. Per a comprovar aquests dos casos es consulta el fitxer *words\_file\_clean.txt*, exactament la línia PS, que és on estan les paraules de prova o paraules que demostren que l'informe és de prova (p.e. PRUEBAS,PACIENTE PRUEBA,NO INTERPRETAR). Si l'informe conté alguna d'aquestes paraules, s'escriu en un fitxer dins de la carpeta *strange*. Ordena les paraules finals (PF) en un vector segons el seu índex en l'informe, de menor a major. Aquest vector conté les paraules des d'on començar a buscar les paraules inicials, ja que algunes vegades al principi de l'informe hi ha paraules com "Dr."

Agafa un índex *begin* del que comença a mirar l'índex *ini*. S'entén *ini* com l'índex del que començarà l'informe net. Ordena les paraules finals (PF) en un vector segons el seu índex dins l'informe, de major a menor, aquesta vegada amb el mètode *rfind* que agafa l'última ocurrència de la paraula. Agafa un índex *fin* que serà l'índex final del nou informe. Ordena les paraules inicials (PI) en un vector segons el seu índex en l'informe, de menor a major. Agafa les paraules inicials i les busca des de l'índex *begin* fins l'índex *fin*, si existeix una, agafa el seu índex i es crea l'índex *ini*.

Per acabar, crea el nou informe des de l'índex *ini* fins l'índex *fin*.

Si el nou informe és molt curt s'escriu en *strange* perquè açò significa que no té informació rellevant. Es crea l'informe des de *ini* a *fin* i després es crida al mètode *clean\_dr()* per a eliminar les possibles dades personals que no s'han evitat. Per últim, s'escriu en la carpeta *clean*.

El format perfecte per a retallar els informes amb els índexs descrits anteriorment és: *Dr. loremipsum [...] Informe clínico [...] Atentamente Dr*

- Índex *begin* seria l'índex de "Dr." del principi.
- Índex *ini* seria l'índex de "Informe clínico".
- Índex *fin* seria l'índex de "Atentamente".

Aleshores es buscaria *ini* a partir de *begin*, i es buscaria *fin* a partir de *ini*, de manera que al tallar de *ini* a *fin*, l'informe quedaria de la següent forma: *Informe clínico [...] Atentamente*

Després en *clean\_dr()* s'eliminarà "Atentamente" i qualsevol nom que quede enmig (si aquest nom està en el fitxer *words\_file\_clean.txt*).

*clean\_dr()*: Busca les paraules a eliminar (PE) i, si apareixen en l'informe ja retallat, es reemplacen per una cadena buida.

*config\_default.py*: Fitxer de configuració amb les variables globals dels scripts. Aquest fitxer conté les variables dels següents àmbits:

- Base de dades
- Paràmetres per a la consulta de la base de dades
- Consultes *MySQL*
- API
- Rutes de fitxers

Aquest mòdul va ser el més costós dels quatre. Va ser el mòdul al què més temps es va haver de dedicar en la seua realització i, a més, el que més problemes va produir. Els problemes van sorgir perquè no tots els informes segueixen una plantilla i d'un mateix client poden hi haver diferents tipus d'informe. Relacionat amb aquest problema, el programa que s'ha de realitzar en aquest projecte ha de poder netejar els informes de qualsevol client i per a cada client també hi han diferents tipus d'informe.

Finalment es va solucionar gràcies al fitxer *words\_file\_clean.txt*, revisant els informes i ficant en aquest fitxer les paraules necessàries per a la neteja. També es va solucionar quan es va implementar que aquestes paraules s'ordenaren abans de cercar i crear els índexs.

#### 4.1.4 Mòdul Càrrega

En aquest mòdul es van crear 4 fitxers:

- *Connection.py*: conté la classe *Connection* per a la connexió amb l'API.
- *InteractionBS.py*: conté la classe *InteractionBS* per a realitzar les cridades a l'API del buscador, utilitza una instància de la classe *Connection*.
- *ObjectBS.py*: conté la classe *ObjectBS* per emmagatzemar el resultat d'una cridada GET a l'API.
- *ObjectPOST.py*: conté la classe *ObjectPOST* per a comprovar el resultat d'una cridada POST a l'API del buscador.



Sobre aquestes classes es realitzaran els tests de la fase següent. D'aquesta manera també es comprova d'una forma més visual que els informes estiguen nets; ja que, una vegada es recull un informe del buscador, es retornaran els següents camps:

- *anat*: camp anatomia de l'informe
- *summary*: camp on es troben els noms dels pacients i doctors
- *description*: contingut de l'informe
- *conclusion*: conclusions de l'informe

Si el camp *summary* està buit significa que no conté dades personals de cap tipus. Aquest mòdul carrega els informes per separat, la càrrega de tots els informes la realitza l'empresa. Aquesta envia un arxiu comprimit amb tots els informes, una vegada ja nets, a l'empresa Semantibots per a que els carregue a l'API del buscador.

Per a que tots aquests programes funcionen per a tots els clients de l'empresa, s'ha realitzat un quart programa que és el *config\_default.py*. En aquest programa estan totes les variables dels anteriors programes i així es pot canviar fàcilment la URL (*Uniform Resource Locator*) de la cridada a l'API, les carpetes on guardar els fitxer, i altres variables dels programes.

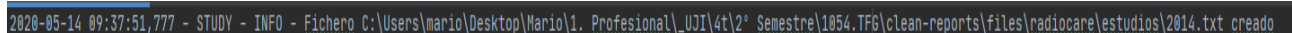
#### 4.1.5 Fitxers .log

Els fitxers *log* son una part important del projecte, ja que es mostra per la terminal tota l'execució del programes. D'aquesta manera ajuden a comprovar que els programes estan funcionant com deuriem i, a banda, queda registrat en els fitxers per a tindre un historial d'errors comprensible per a tota l'empresa.

Aquests fitxers es creen cada 24 hores ja que estan configurats gràcies a un fitxer *log\_config.conf*. Existeixen 3 *logs*, és a dir, es creen 3 fitxers cada dia amb noms diferents. Cada dia es crea un però els que ja estan creats no s'eliminen, sinó que canvien el seu nom: es fica la data del dia que han sigut creats, per exemple, de *readings\_study.log* es canviaria a *readings\_study.log.2020-05-14* si aquest ha sigut creat en aquesta data. Els *logs* del projecte són els següents:

- *readings\_study.log*
- *readings\_feed.log*
- *readings\_clean.log*

***readings\_study.log***: En el primer de tots, es registra l'any dels estudis i la ruta on s'ha creat el fitxer que conté els estudis. Exemple de fitxer *readings\_study.log* en la Figura 4.1.



```
2020-05-14 09:37:51,777 - STUDY - INFO - Fichero C:\Users\mario\Desktop\Mario\1. Profesional\UJI\4t\2º Semestre\1054.TFG\clean-reports\files\radiocare\estudios\2014.txt creado
```

Figura 4.1: *readings\_study.log*.

***readings\_feed.log***: En el fitxer *readings\_feed.log* es registra l'any dels estudis, la URL on s'agafen els informes i si aquests estudis tenen informe (*false* si no té informe, *true* si en té). Exemple de fitxer *readings\_feed.log* en la Figura 4.2 <sup>1</sup>.

```
2020-05-14 09:44:20,006 - FEED - INFO - 2014 | https://actualpacs.com/actualpacs/api/report/plainText/2.16.840.1.113662.2.1.1722.11379.1140109.1093508 | False | 193
2020-05-14 09:44:21,744 - FEED - INFO - 2014 | https://actualpacs.com/actualpacs/api/report/plainText/1.2.840.113619.2.134.1762805279.2197.1389277810.366 | False | 194
2020-05-14 09:44:23,276 - FEED - INFO - 2014 | https://actualpacs.com/actualpacs/api/report/plainText/1.3.51.0.7.11656259708.35857.1860.35136.68306.7878.43003 | False | 195
2020-05-14 09:44:25,125 - FEED - INFO - 2014 | https://actualpacs.com/actualpacs/api/report/plainText/1.3.51.0.7.13776947625.7280.4680.43584.19046.48617.16279 | False | 196
2020-05-14 09:44:26,866 - FEED - INFO - 2014 | https://actualpacs.com/actualpacs/api/report/plainText/1.2.840.113564.3.1.2.192.168.1.30.201401099085632983080 | False | 197
2020-05-14 09:44:28,824 - FEED - INFO - 2014 | https://actualpacs.com/actualpacs/api/report/plainText/1.3.51.0.7.1322988594.15544.47948.38816.45065.4906.63241 | False | 198
```

Figura 4.2: *readings\_feed.log*.

***readings\_clean.log***: En el fitxer *readings\_clean.log*, es registra el *study\_iuid* del informes, el número d'aquest informe segons la seua posició en el fitxer. També es registra la paraula CLEAN si aquest informe s'ha netejat bé i va a al fitxer de la carpeta *clean*, o STRANGE si és de prova o no té informació i va al fitxer de la carpeta *strange*. Exemple de fitxer en la Figura 4.3.

```
2020-05-13 09:22:18,552 - CLEAN - INFO - CLEAN: 1.2.392.200036.9116.2.1.1.8812167.344
2020-05-13 09:22:18,554 - CLEAN - INFO - STRANGE: 1.3.51.0.7.3919119106.15392.2124.38942.18011.690.37588
2020-05-13 09:22:18,561 - CLEAN - INFO - CLEAN: 1.2.840.113619.2.81.290.1.2317.20150106.273209
2020-05-13 09:22:18,567 - CLEAN - INFO - CLEAN: 1.2.840.113619.2.55.3.2831155456.374.1420553786.495
2020-05-13 09:22:18,572 - CLEAN - INFO - CLEAN: 1.2.840.113619.1.20.1126117155120124000.20150106.8470
2020-05-13 09:22:18,577 - CLEAN - INFO - CLEAN: 1.2.840.113619.1.20.1126117155120124000.20150106.8469
2020-05-13 09:22:18,583 - CLEAN - INFO - CLEAN: 1.2.840.113619.2.55.3.2831155456.374.1420553786.566
2020-05-13 09:22:18,587 - CLEAN - INFO - CLEAN: 1.2.840.113564.3.1.2.192.168.1.30.20150106173748890890
2020-05-13 09:22:18,591 - CLEAN - INFO - CLEAN: 1.3.51.0.7.13636882800.46824.20244.43925.50143.62071.54457
2020-05-13 09:22:18,596 - CLEAN - INFO - CLEAN: 1.2.840.113742.1002.112463.11.781.11150105.1162651
```

Figura 4.3: *readings\_clean.log*.

<sup>1</sup>Les URLs/accessos estan modificats per a guardar la privacitat amb el client. També s'han ocultat contrasenyes i noms d'usuari.

### 4.1.6 Frontend

En aquesta fase de la implementació es va realitzar un projecte a banda amb *Django* per a crear aquest projecte i crear la funcionalitat per controlar les vistes; i HTML per a crear aquestes vistes. També es va utilitzar *Ajax* per a connectar les vistes d'HTML i la funcionalitat en *Django* i poder actualitzar les dades més dinàmicament.

L'objectiu d'aquesta fase és crear una demostració per a mostrar el treball realitzat en la presentació d'aquest treball de final de grau. En aquesta demostració s'han realitzat dos vistes: de *login* i del buscador.

#### 4.1.6.1 Vista Login

En aquesta vista (Figura 4.4) s'ha creat un formulari on ficar el nom d'usuari i la contrasenya. S'omplin els dos camps i al pulsar el botó es canvia a la vista del buscador.

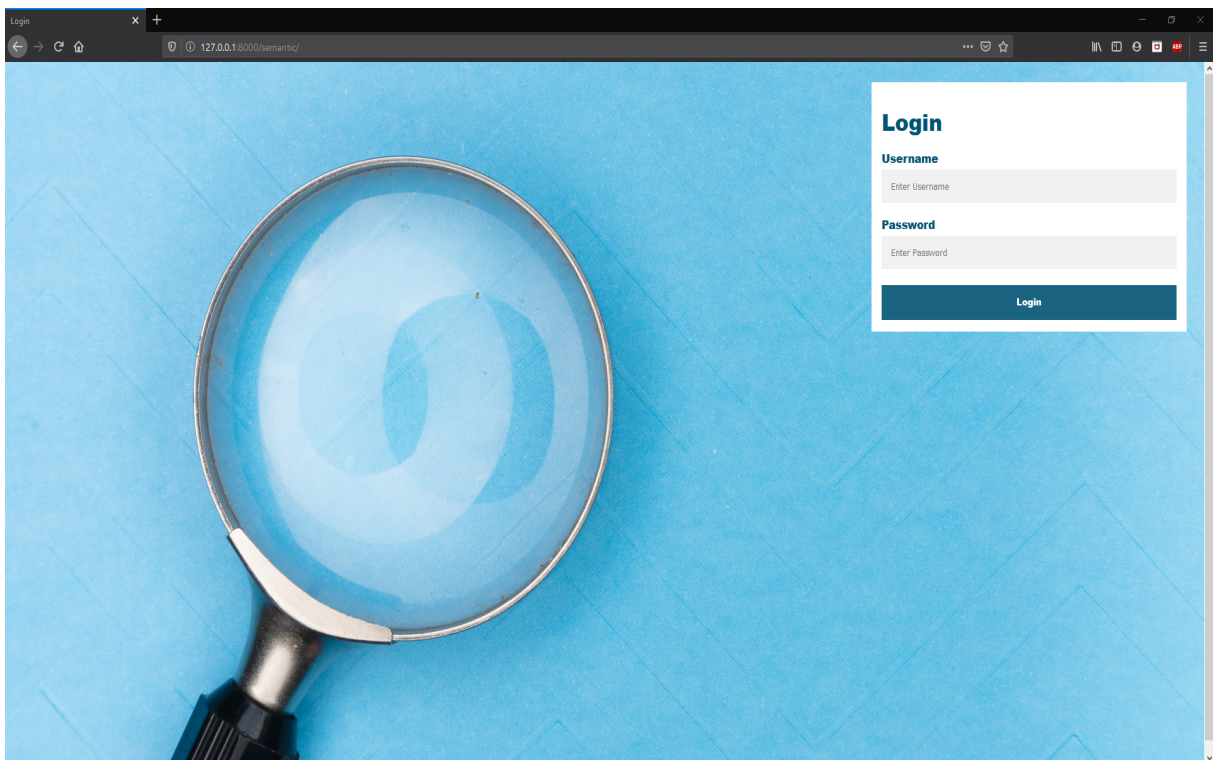


Figura 4.4: Login.

#### 4.1.6.2 Vista Buscador

El buscador (Figura 4.5) realitza una cridada GET amb *Ajax* a la part del *backend* passant-li la paraula a cercar i el valor del botó "Search reports" o "Search articles". En el *backend*, segons

el botó que s'hage polsat, farà una cridada a l'API del buscador per a cercar informes o articles que tinguen eixa paraula.

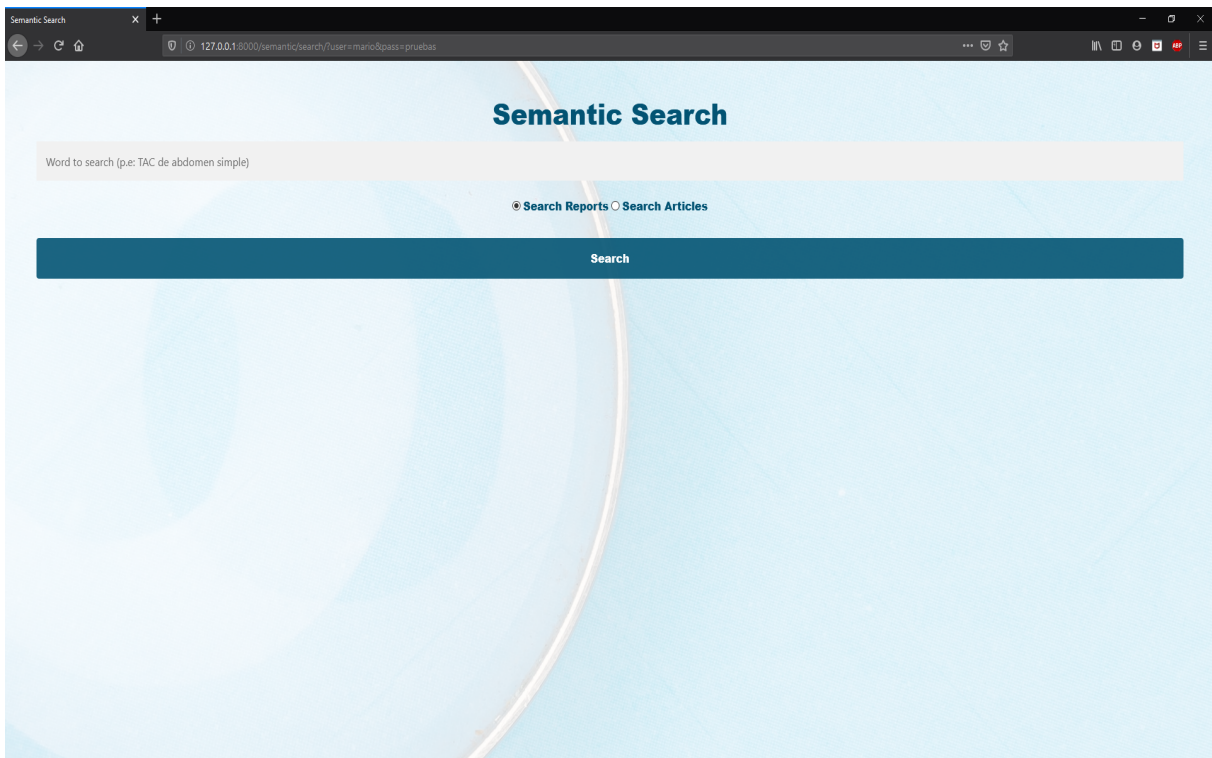


Figura 4.5: Buscador Semàntic.

Per mostrar les dades dels informes i dels articles es va utilitzar un element *d'Ajax*, les *DataTables*. Amb les *DataTables* es van poder crear taules d'informes i d'articles amb paginació i de actualització de dades molt dinàmica (Figura 4.6). També és va facilitar la creació de botons per a importar les dades a *csv* o *Excel*, una vegada realitzat una cerca.

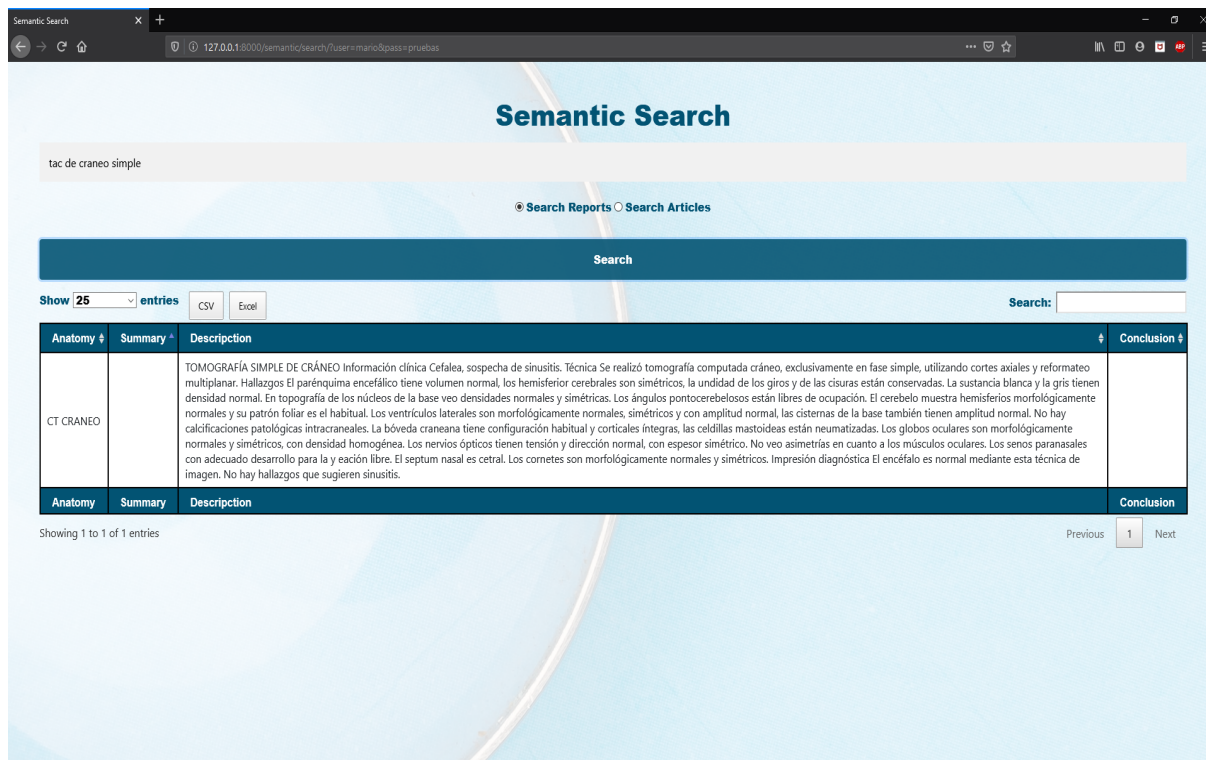


Figura 4.6: Buscador Semàntic (Datatables).

Aquest segon projecte es va realitzar per a tindre una demostració del buscador semàntic, ja implementat en la versió quarta d'ActualPacs. D'aquesta manera, a la presentació d'aquest treball de final de grau, es poden mostrar els informes que s'han pujat a l'API del buscador ja nets de dades personals.

Les anteriors imatges mostren el disseny de les vistes d'aquesta fase. Es compren que han canviat des de la fase de disseny d'aquest projecte degut a les tecnologies i el canvi de funcionament d'aquesta demostració al final del desenvolupament del projecte.

## 4.2 Verificació y validació

En aquesta fase es van realitzar el codi necessari per a provar el funcionament del buscador semàntic. Es van crear proves d'integració i unitaris per a realitzar aquesta comprovació.

### 4.2.1 Test d'integració

Els tests d'integració realitzats en aquest projecte són per a comprovar la connexió de l'API del buscador semàntic amb el mòdul de càrrega explicat en l'apartat d'implementació. Aquests tests estan dins del fitxer de Python `test_connection_bs.py`.

Es van realitzar proves de casos vàlids i invàlids com es pot veure en els apartats següents. D'aquesta manera, si es canvia alguna funcionalitat, des de la part de l'API sabrem detectar l'error i corregir-lo.

#### 4.2.1.1 test\_connection\_bs.py

En aquest fitxer es van crear les proves per a comprovar la connexió amb l'API del buscador semàntic. Es van realitzar dos proves per a comprovar el codi de la resposta http, i tant en el cas vàlid com en el invàlid el codi de la resposta sempre és 200, però en el de la connexió invàlida el JSON que mostra la web conté la paraula error. Açò es pot observar a la Figura 4.7 i Figura 4.8.

```

import unittest
from config import config_default
from src.main.classes.Connection import Connection
import json

URL_TEST = config_default.URL_TEST
URL_POST = config_default.URL_POST
HEADERS = config_default.HEADERS
CENTRO = 'test'
WORD = 'Tomografía'

class MyTestBS(unittest.TestCase):

    def setUp(self) -> None:
        self.connection = Connection(URL_TEST, HEADERS)
        self.texto = "1.2.840.113619.2.22.287.1.6985.20141231.289728 | CT | | Información clínica Paciente con ictericia obstructiva. Técnica Se realiza TAC de abdomen superior con corte hechos cada 10.0 mm sin \
material de contraste oral o IV observándose lo siguiente: Hallazgos En los cortes más superiores se identifican ambas bases pulmonares sin evidencia de infiltrados, consolidaciones o \
derrame pleural. El hígado es de forma, tamaño y densidad normal, sin evidencia de lesiones focales, mencionándose discreta dilatación de la vía biliar intrahepáticas. La vesícula \
biliar es de dimensiones normales sin evidencia de litos en su interior y de pared gada. El páncreas es de aspecto normal, sin embargo debido al grosor corte pequeñas lesiones. \
pueden pasar inadvertidas. El bazo y glándulas suprarrenales son de aspecto normal. Ambos riñones son de forma, tamaño y densidad normal, sin evidencia de lesiones focales. 1 \
litos o hidronefrosis. El trayecto y calibre de ambos ureteres son normales. Las estructuras tubo digestivo observadas no mostraron alteración. No hay evidencia de líquido \
libre en cavidad peritoneal. No se observan crecimientos ganglionares mesentéricos ni retroperitoneales. Impresión diagnóstica 1. Discreta dilatación de la vía biliar. \
intrahepáticas. 2. Se sugiere complementar con estudio con contraste intravenoso y cortes finos cada 3.0 mm a nivel de la encrucijada pancreatobiliar para una mejor valoración."

    def test_connection(self):
        self.connection.set_param(WORD)
        self.assertEqual(self.connection.get_status_get(), 200)

    def test_invalid_connection(self):
        self.connection.set_param('')
        self.assertEqual(self.connection.get_status_get(), 200)

    def test_invalid_connection_data(self):
        self.connection.set_param('')
        data = self.connection.get()
        self.assertNotEqual(data['error'], '')

```

Figura 4.7: test\_connection(1).py.

```
def test_invalid_connection_data(self):
    self.connection.set_param('')
    data = self.connection.get()
    self.assertNotEqual(data['error'], '')

def test_valid_connection_post(self):
    data = json.dumps({"cent": CENTRO,
                      "texto": self.texto})
    self.connection.set_url(URL_POST)
    self.connection.set_param('')
    self.assertEqual(self.connection.get_status_post(data), 200)

def test_connection_post_invalid_format(self):
    response_result = json.dumps({'error': 'Formato erroneo, falta clave centro o texto'})
    data = json.dumps({"cent": CENTRO,
                      "texto": self.texto})
    self.connection.set_url(URL_POST)
    self.connection.set_param('')
    response = self.connection.post(data)
    self.assertEqual(json.loads(response_result), response)
```

Figura 4.8: test\_connection(2).py.

**Problemes:** Un dels problemes que va sorgir durant l'implementació d'aquests test és que el manual de l'API no mostra els nous canvis que ha sofrit l'API i alguns missatges d'errors s'han obtingut al fer proves manuals.

## 4.2.2 Tests unitaris

Els tests unitaris que es van realitzar en aquesta fase són per la comprovació de distintes funcionalitats del codi; la creació dels directoris i fitxers *logs*; els codis HTTP (*Hypertext Transfer Protocol*) i les respostes dels mètodes GET i POST amb l'API del buscador.

### 4.2.2.1 test\_create\_logs.py

En aquest fitxer de tests es comprova si a l'executar qualsevol mètode dels programes es creen els fitxers *logs* relacionats amb cada un d'aquests programes, i es crea en el lloc adequat i amb el nom adequat. Aquests tests es mostren a la Figura 4.9.

```

import unittest
from src.main.classify_study import create_dir_log, dir_comprobatation
import src.main.create_feedfile
import src.main.clean_file
import os
from config import config_default as config

LOG_DIR = config.LOG_DIR

class MyTestCase(unittest.TestCase):

    def setUp(self) -> None:
        self.year = 2015

    def test_create_dir_logs_exists(self):
        dir_comprobatation(self.year)
        self.assertTrue(os.path.exists(LOG_DIR))

    def test_create_readings_study(self):
        dir_comprobatation(self.year)
        self.assertTrue(os.path.exists(os.path.join(LOG_DIR, 'readings_study.log')))

    def test_create_readings_feed(self):
        self.assertTrue(os.path.exists(os.path.join(LOG_DIR, 'readings_feed.log')))

    def test_create_readings_clean(self):
        self.assertTrue(os.path.exists(os.path.join(LOG_DIR, 'readings_clean.log')))

```

Figura 4.9: test\_create\_logs.py.

**Problemes:** El principal problema que va sorgir va ser crear rutes en el fitxer *config\_default.py* que funcionaren des de l'IDE i des de la terminal.

**Solucions:** Una de les solucions que vaig trobar i em va funcionar va ser utilitzar la llibreria *pathlib* de Python en el fitxer *config\_default.py* d'aquesta manera:

```

from pathlib import Path BASE_PATH = Path(os.path.dirname(os.path.realpath(__file__))).parent

```

Al realitzar aquests canvis, els tests passen tots i es creen les carpetes i els fitxers *.log* en el lloc adequat.

#### 4.2.2.2 test\_object\_bs.py

Aquestes proves es van implementar amb l'objectiu de saber que torna i com torna les dades l'API al cercar una paraula. Es van implementar tant casos vàlids com invàlids per a la cerca d'informes i articles. Es pot observar a la Figura 4.10, Figura 4.11 i Figura 4.12.



```

import unittest
from src.main.classes.InteractionBS import InteractionBS
from src.main.classes.ObjectBS import ObjectBS
from src.main.classes.ObjectPOST import ObjectPOST

CENTRO = 'test'

class TestObjectBS(unittest.TestCase):

    def setUp(self) -> None:
        self.valid_word = "TAC de abdomen superior"
        self.invalid_word = "if"
        self.invalid_param = ""
        self.interaction = InteractionBS()
        self.texto = "1,2,840,113619,2,22,287,1,6985,20141231,288728 | CT | | Información clínica Paciente con ictericia obstructiva. Técnica Se realiza TAC de abdomen superior con corte hechos cada 10.0 mm sin " \
                    "material de contraste oral o IV observándose lo siguiente: Hallazgos En los cortes más superiores se identifican ambas bases pulmonares sin evidencia de infiltrados consolidaciones o " \
                    "derrame pleural. El hígado es de forma, tamaño y densidad normal, sin evidencia de lesiones focales, mencionándose discreta dilatación de la vía biliar intrahepáticas. La vesícula " \
                    "biliar es de dimensiones normales sin evidencia de litos en su interior y de pared gada. El páncreas es de aspecto normal, sin embargo debido al grosor, corte pequeñas lesiones " \
                    "quedan pasar inadvertidas. El bazo y glándulas suprarrenales son de aspecto normal. Ambos riñones son de forma, tamaño y densidad normal, sin evidencia de lesiones focales. 1 " \
                    "litos o hidronefrosis. El trayecto y calibre de ambos ureteres son normales. Las estructuras tubo digestivo observadas no mostraron alteración. No hay evidencia de líquido " \
                    "libre en cavidad peritoneal. No se observan crecimientos ganglionares mesentéricos ni retroperitoneales. Impresión diagnóstica 1. Discreta dilatación de la vía biliar. " \
                    "intrahepáticas. 2. Se sugiere complementar con estudio con contraste intravenoso y cortes finos cada 3.0 mm a nivel de la encrucijada pancreatobiliar para una mejor valoración."

    def test_get_search(self):
        object_bs = ObjectBS(self.interaction.get_search(self.valid_word))
        self.assertIsNotNone(object_bs)

    def test_get_search_void(self):
        object_bs = ObjectBS(self.interaction.get_search(self.invalid_word))
        self.assertIsNotNone(object_bs)

    def test_get_search_error(self):
        object_bs = ObjectBS(self.interaction.get_search(self.invalid_param))
        self.assertIn('Error', object_bs._str_())

```

Figura 4.10: test\_object\_bs.py.

```

def test_format_get_search(self):
    data = self.interaction.get_search(self.valid_word)
    self.assertIn('Queries', data)
    self.assertIn('reports', data['Queries'])
    self.assertIn('articles', data['Queries'])
    self.assertIn('global_ranking', data['Queries']['reports'])
    self.assertIn('global_ranking', data['Queries']['articles'])

def test_format_get_search_void(self):
    data = self.interaction.get_search(self.invalid_word)
    self.assertIn('Queries', data)
    self.assertIn('reports', data['Queries'])
    self.assertIn('articles', data['Queries'])
    self.assertIn('global_ranking', data['Queries']['reports'])
    self.assertIn('global_ranking', data['Queries']['articles'])

def test_get_list_reports_no_void(self):
    data = self.interaction.get_list_reports(self.valid_word)
    self.assertNotEqual(data, [])

def test_get_list_reports__void(self):
    data = self.interaction.get_list_reports(self.invalid_word)
    self.assertEqual(data, [])

def test_get_list_reports_data(self):
    data = self.interaction.get_list_reports(self.valid_word)
    object_bs = ObjectBS(self.interaction.get_search(self.valid_word))
    self.assertEqual(data, object_bs.get_reports())

def test_get_list_articles_no_void(self):
    data = self.interaction.get_list_articles(self.valid_word)
    self.assertNotEqual(data, [])

def test_get_list_articles_void(self):
    data = self.interaction.get_list_articles(self.invalid_word)
    self.assertEqual(data, [])

```

Figura 4.11: test\_object\_bs(2).py.

```

def test_get_list_articles_data(self):
    data = self.interaction.get_list_articles(self.valid_word)
    object_bs = ObjectBS(self.interaction.get_search(self.valid_word))
    self.assertEqual(data, object_bs.get_articles())

def test_format_get_reports(self):
    report = self.interaction.get_reports(self.valid_word)
    self.assertIn("anat", report[0])
    self.assertIn("summary", report[0])
    self.assertIn("description", report[0])
    self.assertIn("conclusion", report[0])
    self.assertIsNotNone(report[0]["anat"])
    self.assertIsNotNone(report[0]["summary"])
    self.assertIsNotNone(report[0]["description"])
    self.assertIsNotNone(report[0]["conclusion"])

def test_invalid_format_get_reports(self):
    self.interaction.connection_get.set_url("https://[REDACTED]/actualmed/search/x?informe_id=")
    report = self.interaction.get_reports(self.invalid_word)
    self.assertIsNone(report)

def test_format_get_articles(self):
    report = self.interaction.get_articles(self.valid_word)
    self.assertIn("title", report[0])
    self.assertIn("abstract", report[0])
    self.assertIn("url", report[0])
    self.assertIsNotNone(report[0]["title"])
    self.assertIsNotNone(report[0]["abstract"])
    self.assertIsNotNone(report[0]["url"])

def test_invalid_format_get_articles(self):
    self.interaction.connection_get.set_url("https://[REDACTED]/actualmed/search/test?informe_id=")
    report = self.interaction.get_articles(self.invalid_word)
    self.assertIsNone(report)

def test_post_and_get(self):
    report = self.interaction.get_list_reports(self.valid_word)
    report_text = self.texto.split(' | ')
    self.assertNotIn(report_text[0], report)

```

Figura 4.12: test\_object\_bs(3).py.

**Problemes:** Els problemes van sorgir quan es va implementar la funcionalitat d'indexar informes, és a dir, de pujar informes nets, que es van netejar amb el programa d'aquest projecte, a l'API.

- Un dels problemes era que el text de l'informe es pujava al camp "summary" però aquest camp és per a les dades personals del pacient i del doctor, així que en aquest cas hauria d'estar en blanc.
- I l'altre problema va ser que quan cercava una paraula que tenia aquest informe, no apareixia l'informe en la llista d'informes que tornava l'API.

#### Solucions:

- Per al primer problema, l'empresa de l'API del buscador semàntic va haver de canviar alguna funcionalitat per a que els informes que es pujaven des d'aquest programa s'indexaren de forma correcta.
- Pel que respecta al segon, no és un problema. El que passa és que tots els informes tenen un índex d'importància i són els que es mostren quan es realitza una cerca. Però aquests nous informes no tenen l'índex d'importància així que per això no apareixen.

Sí que es poden recollir aquests informes si en comptes de realitzar la cerca donant una paraula, es busca per el *study\_ivuid*. D'aquesta manera es va poder comprovar, que els informes que es pujaven des d'aquest programa, s'indexaven correctament.

### 4.2.2.3 test\_objectPOST.py

Les proves d'aquest fitxer es van implementar per a comprovar la funcionalitat d'indexar informes al l'API del buscador i així identificar qualsevol nou canvi de l'API. Les proves es mostren a la Figura 4.13 i Figura 4.14.

```

from unittest import TestCase
import config.config_default as config
import json

from src.main.classes.InteractionBS import InteractionBS
from src.main.classes.ObjectPOST import ObjectPOST

URL_POST = config.URL_POST
CENTRO = 'test'

class TestObjectPOST(TestCase):

    def setUp(self) -> None:
        self.centro = "test"
        self.texto = "1.2.840.113619.2.55.3.2831159456.389.1420035563.962 | CT | ABDOMEN | Información clínica.Descartar absceso intraabdominal. Técnica.Se realiza TAC de abdomen." \
                    "completo con corte hechos cada 7.5 mm sin material de contraste oral o IV observándose lo siguiente: Hallazgos En los cortes más superiores se." \
                    "identifican ambas bases pulmonares sin evidencia de infiltrados, consolidaciones o derrame pleural. El hígado es de forma, tamaño y densidad normal." \
                    "sin evidencia de lesiones focales o dilatación de vías biliares intra o extra hepáticas. La vesícula biliar es de dimensiones normales sin evidencia de." \
                    "litos en su interior y de pared gada. El bazo, páncreas y glándulas suprarrenales son de aspecto normal. Ambos riñones son de forma y densidad normal." \
                    "sin evidencia de lesiones focales, litos o hidronefrosis, mencionándose únicamente discreta nefromegalia derecha asociado reticulación de la grasa pericrenal." \
                    "sugestivo de proceso inflamatorio. El trayecto y calibre de ambos ureteres son normales. Se observan dos pequeñas calcificaciones hacia la derecha hueco pélvico." \
                    "fuera trayecto uretero. Las estructuras tubo digestivo observadas no mostraron alteración. No hay evidencia de líquido libre en cavidad peritoneal." \
                    "No se observan crecimientos ganglionares mesentéricos ni retroperitoneales. Útero y anexos de aspecto normal mediante este método de imagen." \
                    "Vejiga llena y de pared gada. Impresión diagnóstica 1. Hallazgos sugestivos de pielonefritis aguda derecha. 2. Favor de correlacionar con datos clínicos."
        self.year = '2015'
        self.interaction_valid = InteractionBS()
        self.word = 'TAC de abdomen complet'

    def test_format_object_post(self):
        data = json.dumps({"centro": self.centro,
                           "texto": self.texto})
        object_post = ObjectPOST(self.centro, self.texto)
        self.assertEqual(data, object_post.__str__())

    def test_post_ok(self): # pruban solo 1 vez
        num = 25
        response = self.interaction_valid.post_report(num, self.year) # despues de probarlo, cambia el parametro num
        response_result = {'response': 'OK'}

```

Figura 4.13: test\_objectPOST.py.

```

def test_post_ok(self): # probar solo 1 vez
    num = 25
    response = self.interaction_valid.post_report(num, self.year) # despues de probarlo, cambia el parametro num
    response_result = {'response': 'OK'}

    self.assertEqual(response, response_result)

def test_post_repeat(self):
    response = self.interaction_valid.post_report(0, self.year)
    response_result = {"error": "document ID already exists"}

    self.assertEqual(response, response_result)

def test_post_get_report_id(self):
    num = 1
    r = self.interaction_valid.post_report(num, self.year) # despues de probarlo, cambia el parametro num
    response_result = {"error": "document ID already exists"}

    self.assertEqual(r, response_result)
    report = self.texto.split(' | ')
    id = report[0]
    reports = self.interaction_valid.get_report_id(id)

    self.assertIn(report[1], reports[0]['anat'])
    self.assertIn(report[2], reports[0]['anat'])
    self.assertEqual(report[3], reports[0]['description'])
    print(reports)

def test_post_get_report_word(self):
    num = 1
    r = self.interaction_valid.post_report(num, self.year) # despues de probarlo, cambia el parametro num
    response_result = {"error": "document ID already exists"}

    self.assertEqual(r, response_result)
    report = self.texto.split(' | ')
    id = report[0]
    reports = self.interaction_valid.get_list_reports(self.word)

    self.assertNotIn(id, reports)

```

Figura 4.14: test\_objectPOST(2).py.

**Problemes:** Un dels problemes que van sorgir, va ser que el mètode que prova d'indexar un informe només funciona una vegada per cada informe. Això passa perquè al pujar un informe que ja està en l'API el missatge del JSON (*JavaScript Object Notation*) canvia i dona "error: document ID already exists".

**Solucions:** La solució va ser insertar un comentari per a que, cada vegada que s'execute aquest mètode, es canvie la variable *num* que fa referència al número de l'informe en el fitxer.

## Capítol 5

# Conclusions

Per acabar aquest document vaig a exposar les conclusions sobre el projecte i sobre tot el que he tret d'aquesta experiència de les pràctiques.

El projecte va finalitzar correctament i es van realitzar totes les tasques esmentades en la fase de planificació. Fins i tot va donar temps d'implementar la demostració per a la presentació d'aquest treball. El programa realitzat encara es pot millorar, factor que no suposa un problema, ja que és modificable gràcies al fitxer *config\_default.py* i *words\_file\_clean.txt*.

Relacionat amb l'àmbit formatiu, durant aquests 3 mesos de pràctiques en Actualtec, he après a utilitzar un nou *framework* (entorn de treball o marc de treball) com és Django; un nou llenguatge de programació, Javascript, més concretament, la llibreria d'Ajax; i noves llibreries i funcionament de Python. També he repassat aquest últim llenguatge de programació. A més a més, he après a utilitzar la llibreria de *logs* de Python i em pareix molt útil per a la comprovació i registre del funcionament dels programes.

Respecte a l'experiència en l'empresa, ha sigut molt bona i m'ha agradat poder treballar en un entorn de treball com aquest, encara que ha sigut només per un mes o menys a causa de la situació de la COVID-19. Treballar des de casa no ha sigut el mateix però les reunions de treball en l'empresa han continuat a diari.

I per últim, relacionat amb la meua experiència personal, han sigut uns 3 mesos de treballar en un projecte que m'ha agradat, envoltat de uns companys d'empresa molt bons que no han dubtat en ajudar-me quan he tingut algun problema. La situació de la COVID-19 ha dificultat adquirir més experiència respecte a treballar en un entorn de treball més adequat però encara així m'he sentit còmode.

A més considero que el meu treball ha tingut una doble utilitat. Per una part li ha sigut útil a l'empresa, la qual no ha dubtat en aplicar-lo. Per altra part, m'ha servit a mi mateix per a consolidar els coneixements previs i aprendre'n molts més de nous.



# Bibliografia

- [1] Actualtec S.L.: Fitxa de empresa, Espaitec (última visita 05/06/2020)  
<https://espaitec.uji.es/va/el-parc/empreses-del-parc/actual-med/>
- [2] ActuaMed: Web de la marca de l'empresa (última visita 15/06/2020)  
<http://www.actualmed.com/>
- [3] Semanticbots: Web de l'empresa (última visita 05/06/2020)  
<https://www.semanticbots.com>
- [4] Xampp: Web de l'aplicació (última visita 08/06/2020)  
<https://www.apachefriends.org/es/index.html>
- [5] Django: Web del tutorial del framework (última visita 15/06/2020)  
<https://docs.djangoproject.com/en/3.0/intro/tutorial01/>
- [6] Git: Web de la ferramenta (última visita 25/06/2020)  
<https://git-scm.com/>
- [7] Trello: Web de la ferramenta (última visita 25/06/2020)  
<https://trello.com/>
- [8] Microsoft Project: Web de la ferramenta (última visita 25/06/2020)  
<https://www.microsoft.com/es-es/microsoft-365/project/project-management-software>
- [9] Lucidchart: Web de la ferramenta (última visita 25/06/2020)  
<https://app.lucidchart.com/>
- [10] Redmine: Web de la ferramenta (última visita 25/06/2020)  
<https://www.redmine.org/>
- [11] Figma: Web de la ferramenta (última visita 25/06/2020)  
<https://www.figma.com/>
- [12] Adobe Colors: Web de la ferramenta (última visita 25/06/2020)  
<https://color.adobe.com/es/create/color-wheel>





# Annex A

## Manual d'usuari

L'objectiu d'aquest projecte és recollir informes d'estudis radiològics d'un client que es troben en una base de dades. A través dels identificadors d'aquests estudis (`study_iuid`) es poden identificar i recopilar. De manera que es necessita tindre accés a la base de dades dels estudis dels clients.

En aquest cas traiem els *study\_iuid* d'una taula que es diu *study*.

### A.0.1 Requisites

- Python 3.8
- Xampp o programa similar per a allotjar una base de dades en MySQL

### A.0.2 Instal·lació

1. `git clone ssh://git@xxxx.xxx/mario.arroyo/clean-reports.git`
2. Descarregar i instal·lar *Xampp* per a poder connectar-te amb la base de dades en local.
3. Per a importar la base de dades des de la terminal:
  - (a) `cd xampp`
  - (b) `mysql -h localhost -o root nombd < rutadel.sql`
  - (c) `mysql -h localhost -o root semantic < study.sql`
4. Instal·lar les llibreries del fitxer *requirements.txt* (Figura A.1)
  - (a) `cd clean_reports`
  - (b) `pip install -r requirements.txt`

```
asgiref==3.2.3
certifi==2019.11.28
chardet==3.0.4
config-path==1.0.0
Django==3.0.3
dnspython==1.16.0
idna==2.9
infinite==0.1
mysql-connector-python==8.0.19
protobuf==3.6.1
pytz==2019.3
requests==2.23.0
six==1.14.0
slugify==0.0.1
sqlparse==0.3.1
urllib3==1.25.8
```

Figura A.1: requirements.txt.

Abans d'executar el codi cal editar el fitxer de configuració següent.

### A.0.3 Fitxer de configuració config\_default.py

En aquest fitxer cal editar el camp CLIENT, amb el nom del client del qual es vol recollir els estudis; aquest camp també s'usa per a la cridada a la API del client. Només s'ha de modificar aquest camp cada vegada que es desitja canviar de client. Aquest fitxer es mostra a la Figura A.2 i Figura A.3 <sup>1</sup>.

---

<sup>1</sup>Les URLs/accessos estan modificats per a guardar la privacitat amb el client. També s'han ocultat contrasenyes i noms d'usuari.

```

import os
from pathlib import Path

# Base de dades
DB_HOST = 'localhost'
DB_USER = 'root'
DB_PASS = ''
DB_NAME = 'semantic'

# Parametres per la consulta de la base de dades
NAME_DATABASE = 'study' # nom de la taula de la base de dades
ID = 'study_id' # nom del camp id dels estudis
TIME = 'created_time' # nom del camp temps dels estudis
MOD = 'mods_in_study' # nom del camp modalitat dels estudis
DESC = 'study_desc' # nom del camp descripció dels estudis

# Consultes
STUDY_PER_YEAR = "SELECT " + ID + ", " + TIME + " FROM " + NAME_DATABASE + " WHERE year(" + TIME + ") = %(time)s GROUP BY " + TIME + " ORDER BY " + TIME + " ASC;"
PARAM_STUDY = "SELECT COALESCE(" + MOD + ", '') , COALESCE(" + DESC + ", '') FROM " + NAME_DATABASE + " WHERE " + ID + " = %(id)s;"

# API
CLIENT = [REDACTED] # client del qual es recolliran els estudis i informes i es realitzarà la neteja d'aquests últims
URL = "https://" + CLIENT + ".actualpacs.com/actualpacs/api/report/plainText/"
HEADERS = {
    "User-Agent": "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/54.0.2840.98 "
    "Safari/537.36"
}
USER = [REDACTED] # usuari de la API
PASS = [REDACTED] # contrasenya de la API
KEY = 'report' # key del json a recollir

# Rutes de fitxers
BASE_PATH = Path(os.path.dirname(os.path.realpath(__file__))).parent
FILES_PATH = os.path.join(BASE_PATH, 'files') # direcció del directori files
CLIENT_PATH = os.path.join(FILES_PATH, CLIENT) # direcció del directori amb el nom del client del camp CLIENT
STUDY_DIR = os.path.join(CLIENT_PATH, 'estudios') # direcció del directori estudis dins del directori CLIENT
REPORT_DIR = os.path.join(CLIENT_PATH, 'reports') # direcció del directori reports dins del directori CLIENT
REPORT_CLEAN_DIR = os.path.join(REPORT_DIR, 'clean') # direcció del directori clean dins del directori reports
REPORT_STRANGE_DIR = os.path.join(REPORT_DIR, 'strange') # direcció del directori strange dins del directori reports
STUDY_PATH = os.path.join(STUDY_DIR, '{}.txt') # direcció dels fitxers dels estudis dins del directori estudis
REPORT_PATH = os.path.join(REPORT_DIR, '{}.txt') # direcció dels fitxers dels informes dins del directori report
CLEAN_PATH = os.path.join(REPORT_CLEAN_DIR, '{}.txt') # direcció dels fitxers dels informes nets dins del directori report/clean

```

Figura A.2: config\_default.py.

```

STRANGE_PATH = os.path.join(REPORT_STRANGE_DIR, '{}.txt') # direcció dels fitxers dels informes de prova dins del
# directori report/strange
FILE = os.path.join(BASE_PATH, '{}/{}.txt') # direcció genèrica de qualsevol fitxer per al programa organize-reports.py
CONFIG_LOG_DIR = os.path.join(BASE_PATH, 'config/logging.cfg')
LOG_DIR = os.path.join(BASE_PATH, 'logs') # direcció del directori on es guarden els logs
LOG_PATH = os.path.join(LOG_DIR, 'readings-{}.log') # direcció dels fitxers logs

# Fitxer para clean_file.py
FILE_CLEAN_PATH = os.path.join(BASE_PATH, "config/words_file_clean.txt") # direcció fitxer words-file-clean.txt

# Test
URL_TEST = "https://[REDACTED].es/actualmed/search/{0}/"
URL_REPORT = "https://[REDACTED].es/actualmed/search/{0}?informe_id="
URL_POST = "https://[REDACTED].es/actualmed/index"

```

Figura A.3: config\_default(2).py.

#### A.0.4 Fitxer `words_file_clean.txt`

Totes les paraules i frases que es posen en aquest fitxer han d'estar separades per comes. No importa l'ordre de les paraules/frases, el programa les ordena.

PI,(paraula/frase),(paraula/frase),

PF,

PS,

PE,

//PI = línia de paraules inicials

//PF = línia de paraules finals

//PS = línia de paraules d'informes estranys

//PE = línia de paraules a eliminar

El format perfecte per a retallar els informes amb els índexs descrits anteriorment és:

*Dr. loremipsim [...] Informe clínic [...] Atentamente Dr*

- Índex *begin* seria l'índex de "Dr." del principi.
- Índex *ini* seria l'índex de "Informe clínic".
- Índex *fin* seria l'índex de "Atentamente".

Llavors es buscaria *ini* a partir de *begin*, i es buscaria *fin* a partir de *ini*; de manera que quan es talla de *ini* a *fin*, l'informe quedaria d'aquesta manera: Informe clínic [...] Atentament

Després en *clean\_dr* s'esborren les ocurrencies de la paraula "Atentament" i qualsevol nom que quede enmig (si aquest nom està en el fitxer *words\_file\_clean.txt* en la línia PE).

##### A.0.4.1 Casos d'excepcions

###### Signatura electrònica:

`[(/actualpacsv2/public/images/firma(3).png)`

El que cal fer és copiar aquesta línia i enganxar-la en la línia PE del fitxer *words\_file\_clean.txt*, i tornar a passar el codi perquè s'elimine cada vegada que aparega en un informe.

PE,[(/actualpacsv2/public/images/firma(3).png)

**Nom propi:**

*Dr. Mario Arroyo*

El que cal fer és copiar aquest nom i enganxar-ho en la línia PE del fitxer *words\_file\_clean.txt*, tornar a passar el codi perquè s'elimina cada vegada que aparega en un informe.

PE,[(/actualpacsv2/public/images/firma(3).png),Dr.,Mario,Arroyo

Funciona per a la majoria dels clients:

**PI**,Información clínica,Check up,Estudio:,Estudio,se realiza

**PF**,DR.,Dr.,DRA.,Dra.,DR,Dr,DRA,Dra,Atentament,Atte

**PS**,PRUEBA, PRUEBA PRUEBA, PRUEBAPRUEBA

**PE**,”noms de doctors”,DR.,Dr.,DRA.,Dra.,DR,Dr,DRA,Dra,Atentamente,Atte

## A.0.5 Ordre d'execució

Per als passos 1r i 2n cal encendre el *Xampp* o el programa per a la base de dades. Per a l'execució val la terminal de *Windows* o la terminal de *bash*. També es pot executar des de l'IDE.

### A.0.5.1 `classify_study.py`

Paràmetres: l'any dels estudis que es vol recollir

Exemple:

```
cd clean-reports
```

```
python -m src.main.classify_study
```

*Introdueix un any: 2015*

Resultat de l'execució: En acabar l'execució es crea un directori *files* (si no estava creat), dins d'aquest altre directori amb el nom del client (nom en el camp **CLIENT** del *config\_default.py*) i dins d'aquest altres dos directoris, estudis i reports. També es crea el directori *logs*, on es guardaran els arxius *.log*. Açò es pot observar a la Figura A.4.

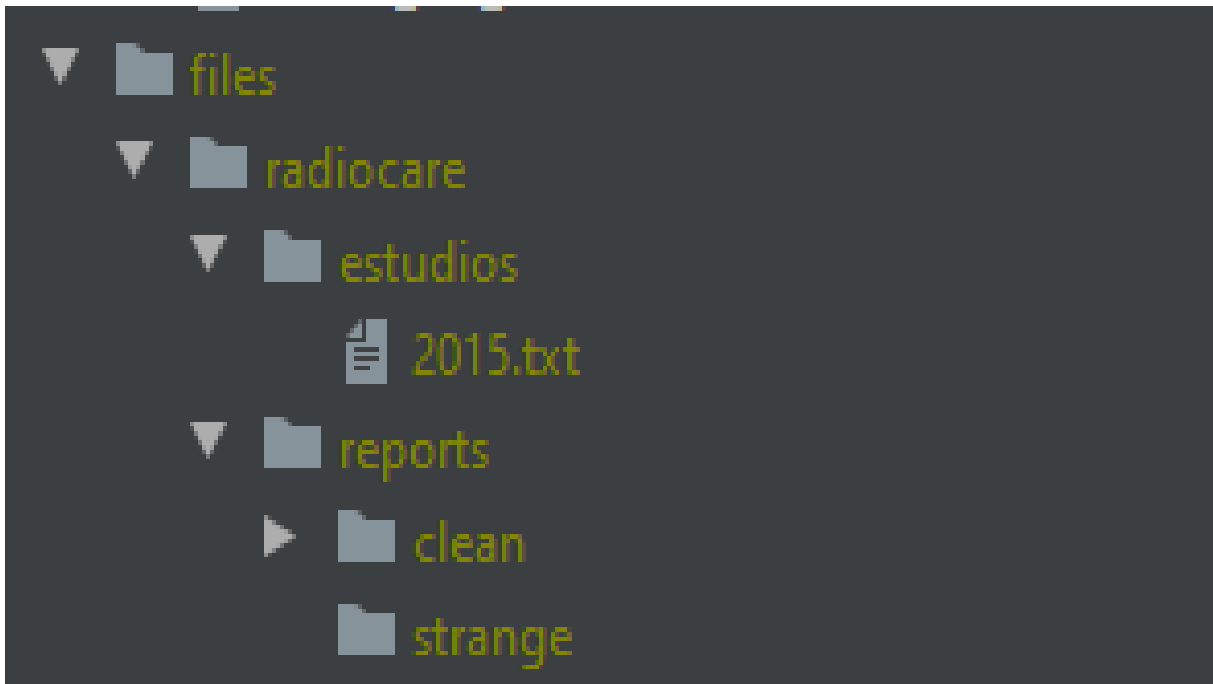


Figura A.4: Resultat `classify_study.py`.

Totes aquestes carpetes s'han creat ara pel mètode `create_directories()`, i la de `reports` i els seus subdirectoris serveixen per als programes següents. Si el fitxer ja existia, s'elimina i es torna a crear. Els directoris `reports`, `reports/clean` i `reports/strange` no s'utilitzen en aquest programa però si en els següents.

#### A.0.5.2 `create_feedfile.py`

Paràmetres:

- L'any dels estudis els informes dels quals vols recollir
- El número de l'estudi que començar a buscar. Aquest número es pot trobar al final de l'última línia del `readings-feed.log`: `2020-05-13 08:52:01,072 - FEED - INFO - 2015 | https://client.actualpacs.com/actualpacs/api/report/plaintext/1.3.51.0.21004 | False | 217`

S'ha d'introduir el número següent a aquest, en aquest cas s'hauria d'introduir el 218.

Exemple:

```
cd clean-reports
```

```
python -m src.main.create_feedfile
```

*Introdueix un any: 2015*

*Introdueix el número de l'últim estudi: 218 (res/0, o número de l'últim estudi)*

En acabar l'execució s'ha creat en reports un fitxer amb els informes dels estudis de l'any introduït, el format del qual és 'study\_iuid | modalitat | descripció | informe'; els primers camps es recullen de la base de dades del client i l'informe de la cridada a la API, i es crea el següent fitxer amb el nom de l'any. Açò es pot observar a la Figura A.5.

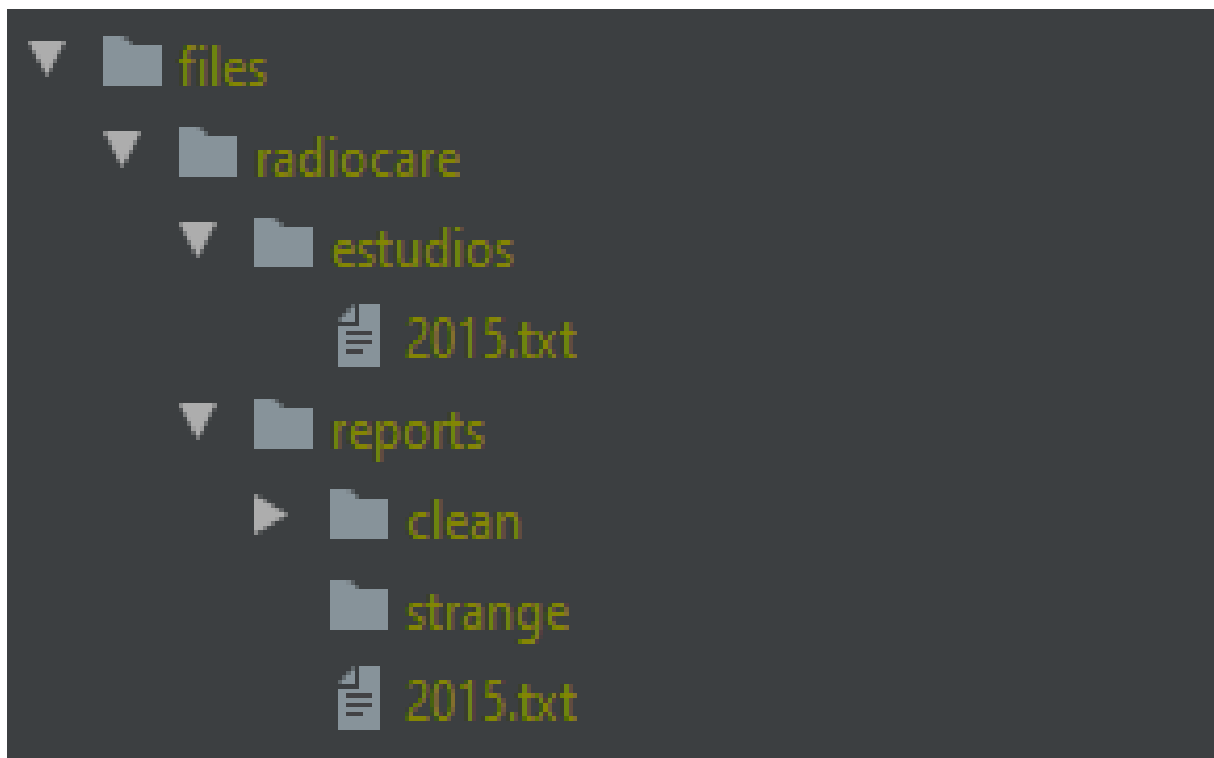


Figura A.5: Resultat create\_feedfile.py.

### A.0.5.3 clean\_file.py

Paràmetres:

- L'any dels informes que vols netejar.
- El número de l'estudi que començar a buscar. Aquest número es pot trobar al final de l'última línia del *readings\_clean.log*:

2020-05-13 08.52:01,072 - FEED - INFO - 2015 |

<https://client.actualpacs.com/actualpacs/api/report/plaintext/1.3.51.0.7.21004> | False | 217

S'ha d'introduir el número següent a aquest, en aquest cas s'hauria d'introduir el 218.

Exemple:

```
cd clean-reports
```

```
python -m src.main.clean_file
```

*Introdueix un any: 2015*

*Introdueix el número de l'últim estudi: 218 (res/0, o número de l'últim estudi)*

En acabar l'execució es crea un fitxer en *reports/clean* amb els informes nets i un altre fitxer en *reports/strange* amb els informes de prova o que no tenen informació (mateix format que el *create\_feedfile*). Açò es pot observar en la Figura A.6.

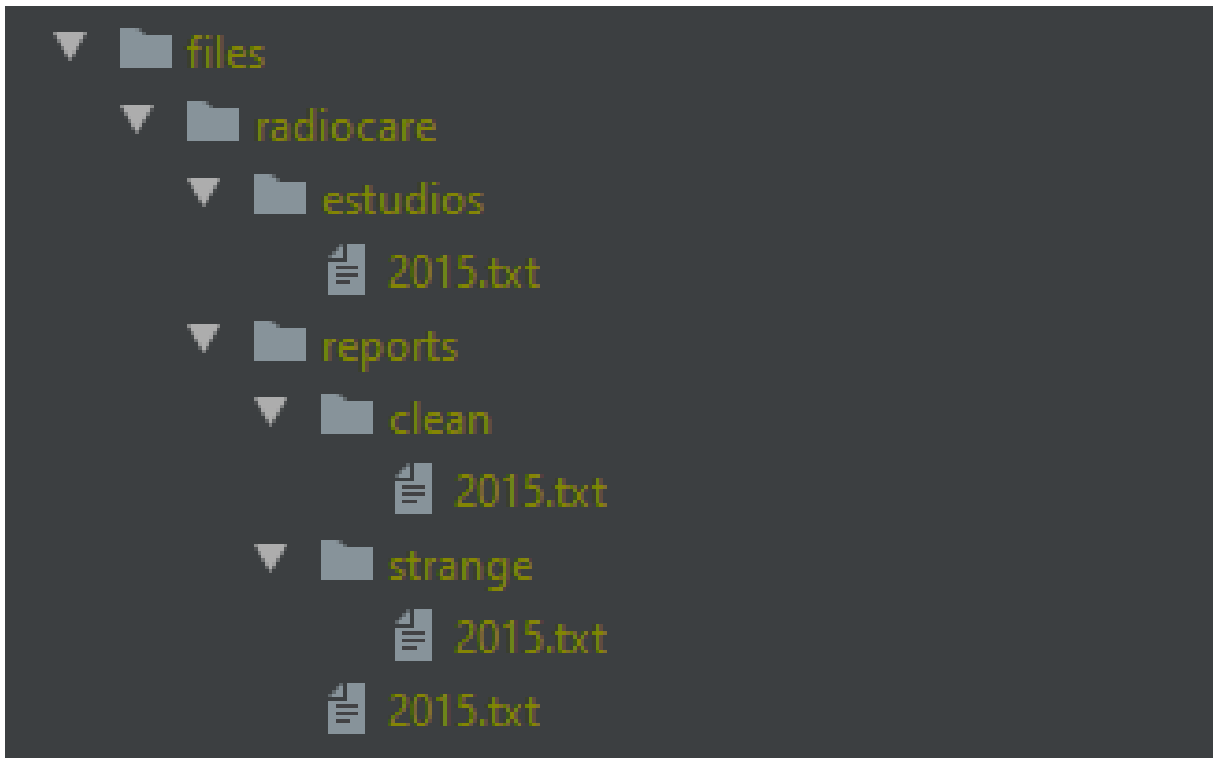


Figura A.6: Resultat `clean_file.py`.

#### A.0.5.4 Execució dels tests

Executar tests des de la consola:

```
python -m unittest discover <test-directory>
```



```
python -m unittest discover src/test
```

El mètode *test\_post\_ok* del fitxer *test\_objectPOST.py* només funciona una vegada. Perquè torne a funcionar s'ha de canviar el número de l'informe que se li passa perquè es faça la indexació d'aquest informe en el cercador.



# Índex de figures

2.1	Diagrama EDT. . . . .	12
2.2	Diagrama Gantt. . . . .	24
3.1	Diagrama de casos d'ús. . . . .	29
3.2	Vista Login. . . . .	34
3.3	Vista Buscador. . . . .	35
4.1	readings_study.log. . . . .	41
4.2	readings_feed.log. . . . .	42
4.3	readings_clean.log. . . . .	42
4.4	Login. . . . .	43
4.5	Buscador Semàntic. . . . .	44
4.6	Buscador Semàntic (Datatables). . . . .	45
4.7	test_connection(1).py. . . . .	46
4.8	test_connection(2).py. . . . .	47
4.9	test_create_logs.py. . . . .	48
4.10	test_object_bs.py. . . . .	49
4.11	test_object_bs(2).py. . . . .	49
4.12	test_object_bs(3).py. . . . .	50
4.13	test_objectPOST.py. . . . .	51

4.14	test_objectPOST(2).py. . . . .	52
A.1	requirements.txt. . . . .	58
A.2	config_default.py. . . . .	59
A.3	config_default(2).py. . . . .	59
A.4	Resultat classify_study.py. . . . .	62
A.5	Resultat create_feedfile.py. . . . .	63
A.6	Resultat clean_file.py. . . . .	64

# Índex de taules

2.1	Diccionari EDT (Introducció) . . . . .	13
2.2	Diccionari EDT (Planificació) . . . . .	14
2.3	Diccionari EDT (Gestió) . . . . .	15
2.4	Diccionari EDT (Anàlisi) . . . . .	16
2.5	Diccionari EDT (Disseny) . . . . .	17
2.6	Diccionari EDT (Implementació) . . . . .	18
2.7	Diccionari EDT (Testeig). . . . .	19
2.8	Diccionari EDT (Tancament i posada en marxa). . . . .	20
2.9	Diccionari EDT (Seguiment i control). . . . .	21
3.1	Especificació cas d'ús (CU01) . . . . .	30
3.2	Especificació cas d'ús (CU02) . . . . .	31
3.3	Especificació cas d'ús (CU03) . . . . .	33