
Desenvolupament d'una aplicació web basada en Machine Learning per analitzar Hashtags i missatges utilitzant bones pràctiques d'enginyeria.

Treball de Fi de Grau
Grau en Enginyeria en Informàtica

Alejandro Gallego Rodríguez

Especialitat: Enginyeria del Software
Director: Silverio Martínez-Fernández
Codirectora: Cristina Gómez Seoane



2021-2022 Q1

Facultat d'Informàtica de Barcelona
Universitat Politècnica de Catalunya

Resum

Cada vegada apareixen noves xarxes socials i amb elles, noves formes de comunicació entre les persones que les utilitzen. Aquestes xarxes socials donen una possibilitat enorme a analitzar grans conjunts de dades extretes dels usuaris de forma directa, sense cap filtre.

En aquest projecte fem servir aquestes dades disponibles per ser analitzades i així fer una anàlisi de sentiment d'un hashtag o una paraula clau de Twitter. Per aconseguir aquest objectiu es desenvolupa una aplicació web amb un component d'aprenentatge automàtic que, entre altres funcionalitats, és capaç d'interpretar i classificar el sentiment d'una oració, en el nostre cas, d'un tweet, sigui positiu, negatiu o neutral. Sent el sentiment d'una frase una mica interpretable o contextualitzable, en alguns casos, el component d'aprenentatge automàtic s'ha desplegat amb una precisió del 80%, classificant així de manera correcta la majoria dels tweets.

Finalment, podem concloure que una eina tan potent com és Twitter ens ofereix multitud d'usos per a la nostra aplicació, ja sigui de forma comercial per part d'empreses que busquen una anàlisi detallada dels seus respectius hashtags, com de forma recreativa per part d'usuaris regulars d'aquesta xarxa social.

Resumen

Hoy en día, cada vez van apareciendo nuevas redes sociales y con ellas, nuevas funcionalidades que innovan las existentes formas de comunicación entre las personas que las utilizan. Estas redes sociales nos dan una posibilidad enorme de analizar grandes conjuntos de datos que son extraídos directamente de los usuarios, sin ningún filtro.

En este proyecto utilizamos esos datos disponibles para ser analizados y así realizar un análisis de sentimiento de un *hashtag* o una palabra clave. Para conseguir este objetivo se desarrolla una aplicación web con un componente de *Machine Learning* que, entre otras funcionalidades, es capaz de interpretar y clasificar el sentimiento de una oración, en nuestro caso, de un *tweet*, ya sea positivo, negativo o neutral. Siendo el sentimiento de una frase algo interpretable o contextualizable, en algunos casos, el componente de *Machine Learning* se ha desarrollado con una precisión del 80%, clasificando así de forma correcta la mayoría de los *tweets*.

Finalmente, podemos concluir que una herramienta tan potente como es *Twitter* nos ofrece multitud de usos para nuestra aplicación, ya sea de forma comercial por parte de empresas que buscan un análisis detallado de sus respectivos *hashtags*, como de forma recreativa por parte de usuarios regulares de esta red social.

Abstract

Nowadays, new social networks are appearing and with them, new functionalities that innovate the existing forms of communication between the people who use them. These social networks give us an enormous possibility to analyze large sets of data that are extracted directly from users, without any filter.

In this project we use that available data to be analyzed and thus carry out a sentiment analysis of a hashtag or a keyword. To achieve this goal, a web application has been developed with a Machine Learning component that, among other functionalities, is capable of interpreting and classifying the sentiment of a sentence, in our case, of a tweet, whether it is positive, negative or neutral. Being the sentiment of a sentence somewhat interpretable or contextualizable, in some cases, the Machine Learning component has been developed with an accuracy of 80%, thus correctly classifying most of the tweets.

Finally, we can conclude that a tool as powerful as *Twitter* offers us a variety of uses for our application, whether commercially by companies seeking a detailed analysis of their respective *hashtags*, or recreationally by regular users of this social network.

ÍNDEX DE CONTINGUTS

1. Introducció	11
1.1. Motivació	11
1.2. Problema a resoldre	12
1.3. Glossari	13
1.4. Actors implicats	15
1.5 Abast del projecte	16
1.5.1. Objectius del projecte	16
1.5.2. Possibles obstacles i riscos	17
1.6. Intel·ligència Artificial	18
1.6.1. Machine Learning	18
1.6.2. Recurrent Neural Network	19
1.6.3. Long-Short Term Memory	20
1.7. Estructura del document	21
2. Justificació	22
2.1. Solucions alternatives	22
2.2. Solució escollida	23
3. Metodologia i rigor	25
3.1. Mètodes de treball	25
3.1.1. Sprint planning	25
3.1.2. Sprint review	25
3.1.3. Sprint retrospective	26
3.1.4 Scrum i Intel·ligència Artificial	26
3.2. Eines de seguiment	28
3.2.1. Trello	28
3.2.2. GitHub	28
3.2.3. Google Drive	29
3.2.4. mllint	29
4. Descripció de les tasques	30
4.1. Planificació del projecte	30

4.2. Incepció del projecte	31
4.3. Primera iteració	31
4.4. Segona iteració	32
4.5. Tercera iteració	33
4.6. Quarta iteració	34
4.7. Quinta iteració	34
4.8. Sexta iteració	35
4.9. Comunicació i memòria.	35
4.10. Recursos	36
4.10.1. Recursos humans	36
4.10.2. Recursos materials.	36
5. Estimacions i diagrama de Gantt	37
5.2. Diagrama de Gantt	38
6. Gestió del risc: Plans alternatius i obstacles	39
7. Pressupost	40
7.1. Identificació dels costos	40
7.1.1. Cost humà	40
7.1.2. Cost software	42
7.1.3. Cost hardware	42
7.1.4. Cost despeses	43
7.2. Estimació dels costos	44
7.3. Control de gestió	45
8. Anàlisi de requisits	46
8.1. Requisits funcionals	46
8.2. Requisits no funcionals	49
8.2.1. Requisits no funcionals d'una aplicació web	49
8.2.2. Requisits no funcionals de components de Machine Learning	51
9. Especificació del sistema	53
9.1. Diagrames de casos d'ús	53
9.1.1. Gestió d'accés al compte	53

9.1.2. Gestió del compte	54
9.1.3. Gestió d'anàlisi de Hashtags	54
9.2. Descripció de casos d'ús	55
9.2.1. Gestió d'accés al compte	55
9.2.2. Gestió del compte	56
9.2.3. Gestió d'anàlisi de Hashtags	57
9.3. Model conceptual	58
9.3.2. Restriccions d'integritat	60
10. Disseny del sistema	61
10.1. Arquitectura	61
10.1.1. Gestió de dades	62
10.1.2. ML Model	62
10.1.3. Capa de model	62
10.1.4. Capa de dades	63
10.1.5. Capa de domini	63
10.1.6. Capa de presentació	63
10.2. Diagrames de seqüència	65
10.3. Disseny de mockups	67
11. Implementació del sistema	69
11.1. Implementació del model	69
11.1.1. Natural Language Processing	69
11.1.2.1. Preprocessament de text	70
11.1.2.2. Tokenització i farciment	70
11.1.2. Entrenament	71
11.1.3. Modelatge i resultats de l'entrenament	72
11.1.4. Ús de mllint al model	75
11.2. Implementació de l'aplicació web	77
11.3. Implementació del servidor	78
11.4. Implementació del servidor Machine Learning	79
12. Validació de requisits	80
12.1. Validació de requisits funcionals	80

12.2. Validació de requisits no funcionals	81
12.2.1. Validació de requisits no funcionals d'aplicacions web	81
12.2.2. Validació de requisits no funcionals de components de ML	82
13. Bones pràctiques d'enginyeria	83
13.1. Controlar el labelling de dades	84
13.2. Eliminar informació redundant	84
13.3. Adoptar una metodologia de software	84
13.4. Assegurar la qualitat en el codi	85
13.5. CI/CD	85
13.6. Realització de tests	85
13.7. Mantenir control de nulls	86
13.8. Divideix i venceràs, reentrenament de model	86
13.9. Utilitzar eines ja existents	86
13.10. Automatitzar l'entrenament del model	87
13.11. Ajustament amb mini-batches	87
14. Discussions	88
14.1. Informe de sostenibilitat	88
14.1.1. Autoavaluació	88
14.1.2. Fita inicial	89
14.1.2.1. Dimensió ambiental	89
14.1.2.2. Dimensió econòmica	89
14.1.2.3. Dimensió social	90
14.1.3. Fita final	90
14.1.3.1. Dimensió ambiental	90
14.1.3.2. Dimensió econòmica	92
14.1.3.3. Dimensió social	93
14.2. Resultats de la planificació	95
14.2.1. Desenvolupament de les iteracions	95
14.2.1.1. Planificació	95
14.2.1.2. Incepció	96
14.2.1.3. Primera iteració	96

14.2.1.4. Segona iteració	97
14.2.1.5. Tercera iteració	98
14.2.1.6. Quarta iteració	99
14.2.1.7. Quinta iteració	99
16.2.1.8. Sexta iteració	100
14.2.2. Costos finals	101
15. Conclusions i treball futur	102
15.1. Conclusions del projecte	102
15.2. Justificació de competències	103
15.3. Integració de coneixements	104
15.4. Treball futur	105
15.5. Valoració personal	106
Bibliografia	107
Annex	112
A. Manual d'usuari	113
A.A. Pantalla principal	113
A.B. Pantalla d'anàlisi de hashtags	114
A.C. Pantalla d'inici de sessió	115
A.D. Pantalla de registre de compte	116
A.E. Pantalla de perfil d'usuari	117
A.F. Pantalla de trending	118
A.G. Pantalla de contacte	119

ÍNDIX DE TAULES

<i>Taula 1: Taula de glossari</i>	14
<i>Taula 2: Comparativa entre solucions</i>	24
<i>Taula 3: Estimacions de les tasques</i>	37
<i>Taula 4: Taula de cost humà</i>	40
<i>Taula 5: Taula de tasques i cost</i>	41
<i>Taula 6: Taula de cost software</i>	42
<i>Taula 7: Taula de cost hardware</i>	43
<i>Taula 8: Taula de cost despeses</i>	43
<i>Taula 9: Taula de contingències</i>	44
<i>Taula 10: Taula de cost imprevistos</i>	44
<i>Taula 11: Taula de pressupost total</i>	45
<i>Taula 12: Taula de validacions de requisits no funcionals d'aplicacions web</i>	81
<i>Taula 13: Taula de validacions de requisits no funcionals de components de ML</i>	82
<i>Taula 14: Taula de costos finals i amortització</i>	101

ÍNDEX DE FIGURES

<i>Figura 1: Flux de treball de sistemes basats en IA</i>	26
<i>Figura 2: Esquema general de l'aplicació</i>	27
<i>Figura 3: Diagrama de Gantt</i>	38
<i>Figura 4: Diagrama casos d'ús de gestió d'accés al compte</i>	53
<i>Figura 5: Diagrama casos d'ús de gestió del compte</i>	54
<i>Figura 6: Diagrama casos d'ús de gestió d'anàlisi de hashtags</i>	54
<i>Figura 7: Esquema conceptual UML</i>	58
<i>Figura 8: Arquitectura del sistema</i>	61
<i>Figura 9: Diagrama de seqüència inici de sessió</i>	65
<i>Figura 10: Diagrama de seqüència cerca de hashtag</i>	66
<i>Figura 11: Mockup cercador i anàlisi de hashtags</i>	67
<i>Figura 12: Mockup pantalla de trendings</i>	68
<i>Figura 13: Flow de la implementació d'un model Machine Learning</i>	69
<i>Figura 14: Distribució d'etiquetes en el dataset</i>	72
<i>Figura 15: Representació de l'algorisme Word2Vec</i>	73
<i>Figura 16: Resum d'arquitectura del model</i>	74
<i>Figura 17: Precisió i perdudes del model en cada iteració</i>	74
<i>Figura 18: Evolució de precisió i perdudes durant l'entrenament</i>	75
<i>Figura 19: Precisió i perdudes resultants del model</i>	75
<i>Figura 20: Captura de pantalla d'un tros d'anàlisi de mllint</i>	76
<i>Figura 21: Entorn de desplegament Azure</i>	77
<i>Figura 22: Documentació d'usuaris del servidor backend</i>	78
<i>Figura 23: Documentació del servidor ML (OpenAPI)</i>	79
<i>Figura 24: Bones pràctiques d'AI Engineering agrupades per categoria</i>	83
<i>Figura 25: Pantalla principal</i>	113
<i>Figura 26: Pantalla d'anàlisi d'un hashtag</i>	114
<i>Figura 27: Pantalla d'inici de sessió</i>	115
<i>Figura 28: Pantalla d'inici de registre de compte</i>	116
<i>Figura 29: Pantalla de perfil d'usuari</i>	117
<i>Figura 39: Pantalla de trending</i>	118
<i>Figura 31: Pantalla de contacte</i>	119

1. Introducció

En aquest capítol es realitza una introducció a la següent memòria, durant aquesta secció es mostrarà la respectiva motivació, es plantejarà el problema a resoldre, un glossari amb els termes més importants i els actors implicats en aquest projecte.

No només això si no que es realitzarà un informe de l'abast d'aquest projecte i s'introduirà els conceptes de *Machine Learning* que s'utilitzaran al llarg d'aquest treball.

1.1. Motivació

El Treball de Fi de Grau "*Desenvolupament d'una aplicació web basada en Machine Learning per analitzar Hashtags utilitzant bones pràctiques d'enginyeria*" pertany al grau d'Enginyeria Informàtica, amb especialitat en Enginyeria del Software en la Facultat d'Informàtica de Barcelona, situada en la Universitat Politècnica de Catalunya.

Durant els anys, s'han anat creant xarxes socials on les persones donen la seva opinió i comparteixen teories. Els mitjans de comunicació, ja sigui programes de televisió com premsa online, han hagut d'adaptar-se a aquestes noves formes de comunicació entre persones, on el *feedback* dels espectadors és molt més directe i transparent. Una de les xarxes socials més grans del món és Twitter, una plataforma impulsada per els seus usuaris on la funcionalitat més important és la publicació de petites opinions de no més de 240 caràcters [21].

Amb l'aparició d'aquesta plataforma l'any 2006 i la introducció d'una actualització per a empreses on es presenta el *hashtag*, Twitter va obtenir un record històric d'usuaris i els programes i empreses van començar a utilitzar-la per rastrejar les opinions del seu públic objectiu, sent així una eina addicional i sense filtres, directament connectada als seus usuaris [22].

Aquest TFG pretén desenvolupar una eina per analitzar un *hashtag* de Twitter d'una forma més personal, fent especial èmfasi en el sentiment de l'usuari que interactua amb el *hashtag*. Durant aquest document s'especifica tota la informació necessària corresponent a l'abast i context, a més de la metodologia que se seguirà en el desenvolupament del projecte.

1.2. Problema a resoldre

Avui dia, podem veure que en tot programa de televisió o esdeveniment online li acompanya un *hashtag*. Un *hashtag* és una paraula o una petita frase identificativa que serveix per centralitzar els comentaris i reaccions sobre aquest programa o esdeveniment i poder mesurar l'impacte d'una manera ordenada.

Els usuaris utilitzen aquest *hashtag* per publicar les seves opinions i poder visualitzar les reaccions de persones que segueixen o *influencers* actius. És per això que és una eina tan potent per mesurar i veure les reaccions del públic.

El problema d'aquesta forma de mesura és que un *hashtag* pot tenir un gran impacte, tant negatiu com positiu. Una empresa pot veure que el seu *hashtag* està en *trending*, però pot ser per motius negatius, donant una falsa impressió de què el seu producte està sent un èxit.

Podem posar el següent exemple, en un programa de televisió publica un *hashtag* (p.e #PrimerPrograma2022) per a que la gent deixi les seves opinions utilitzant-lo. Quan acaba el programa, els redactors d'aquest veuen que el seu *hashtag* és *trending*, és a dir, molta gent de la plataforma ho ha comentat, sent, per tant, un èxit. En el següent programa, esperant una rebuda tan bona com l'anterior, els redactors del programa se sorprenden quan s'adonen que la recepció del segon programa ha sigut mínima.

Els redactors d'aquest programa decideixen accedir a les opinions de l'anterior programa per observar que ha canviat i per què l'actual ha tingut una rebuda tan negativa. A l'hora de indagar en els comentaris amb més impacte s'adonen que els comentaris són negatius i que l'única raó per la qual l'anterior programa havia tingut tant impacte és per motius negatius, donant així una falsa impressió.

Aquest projecte busca resoldre aquest problema, fent un *Sentiment Analysis*. Un *Sentiment Analysis*, com el seu nom indica, és una anàlisi de sentiment. En aquest projecte, per elaborar una anàlisi de sentiment, s'utilitza intel·ligència artificial, més concretament *Machine Learning*, creant un sistema capaç de decidir, per si mateix, si una frase és negativa, positiva o neutral, seguint uns passos que es detallaran en el transcurs d'aquest document [55].

No només això sinó que assegurarem la qualitat de la nostra solució seguint un conjunt de bones pràctiques, tant de disseny com de programació específiques per a sistemes de *Machine Learning* i que s'explicaran més endavant en aquest document [54].

1.3. Glossari

A continuació es mostra la **Taula 1** per tal de facilitar l'enteniment al lector/a d'alguns dels termes que s'utilitzen al llarg d'aquest document.

Terme	Descripció
Sentiment Analysis	Anàlisi de sentiment. Comprovar si una frase té un sentiment positiu, negatiu o neutral.
Twitter	Xarxa social en la qual els usuaris poden publicar petits missatges de no més de 240 caràcters.
Machine Learning	Aprenentatge automàtic. S'utilitzen models per simular prediccions per ordinador.
User-friendly	Bon disseny que ajuda a l'usuari durant la seva visita en la pàgina web i que té facilitat d'ús.
Full-stack developer	Desenvolupador amb habilitat per treballar tant en la part de frontend com en la part de backend.
Frontend	Part visual de la pàgina web, tot allò amb el que un usuari qualsevol pot interactuar directament.
Backend	El backend és la part del desenvolupament web que s'encarrega de què tota la lògica d'una pàgina web funcioni [58].
Metodologies àgils	Metodologies de desenvolupament que es

	basen en l'adaptabilitat de qualsevol canvi com a mitjà per augmentar les possibilitats d'èxit d'un projecte [8].
SCRUM	Marc de treball per desenvolupar, desplegar i gestionar projectes en entorns complexos [6].
Sprint	Esforç durant un període de temps; és a dir, la durada s'acorda i es fixa per endavant per a cada sprint i normalment es troba entre una setmana i un mes [9].
Sprint review	Progrés que es mostra als <i>stakeholders</i> per a que donin un feedback.
Linter	Eina per trobar errors de programació o d'estil, assegurar la qualitat i la seguretat.
Endpoint	Punt d'entrada amb el qual es pot accedir a una RESTful API i utilitzar determinats serveis, depenent del link introduït.
Framework	Entorn de treball. Tecnologia derivada d'un llenguatge de programació o d'una altra tecnologia i que serveix per a un propòsit específic.

Taula 1: Taula de glossari

1.4. Actors implicats

En aquest apartat llistaré els actors implicats o *stakeholders* en el desenvolupament d'aquesta aplicació

Desenvolupador: Com a únic desenvolupador estaré jo mateix, creant i dissenyant la pàgina web, juntament amb les seves respectives funcionalitats.

Director de TFG: Com a director de TFG estarà Silverio Juan Martínez Fernández que ajudarà a guiar el desenvolupament del projecte.

Codirectora de TFG: Com a codirectora de TFG estarà Cristina Gómez Seoane que, juntament amb Silverio, ajudarà a guiar el desenvolupament d'aquest projecte.

Usuaris de Twitter: Els usuaris de familiaritzats amb Twitters podran utilitzar l'aplicació per investigar sobre hashtags que siguin del seu interès o que hagin creat ells mateixos.

Empreses: Aquelles empreses que utilitzen hashtags com a forma de mesura d'impacte sortiran beneficiades en utilitzar aquesta aplicació per poder agafar més informació i reaccions del públic.

1.5 Abast del projecte

Completar aquest projecte suposaria la realització d'una pàgina web amb un component de *Machine Learning* que proporciona a l'usuari una anàlisi d'un *hashtag* o d'una paraula clau que hagi escollit, fent servir bones pràctiques d'enginyeria.

A continuació es detallaran els objectius principals d'aquest projecte per desenvolupar i els possibles riscos que s'han identificat i que poden ocórrer durant el desenvolupament d'aquest treball.

1.5.1. Objectius del projecte

Hi ha una sèrie d'objectius a aconseguir, cadascun d'ells imprescindible per poder desenvolupar aquest projecte de forma exitosa. Els objectius a aconseguir en aquest projecte són els següents:

- **O1: Proporcionar una eina en forma de pàgina web per a tot usuari.** Construir una aplicació web *user-friendly* que tot usuari pugui utilitzar sense cap tutorial.
- **O2: Ajudar a empreses a crear els seus informes en un aspecte emocional de les reaccions.** Eliminar el concepte "Com major impacte millor" i substituir-lo per "Com major impacte positiu millor".
- **O3: Oferir una plataforma amb integració d'un model de Machine Learning.** Aprendre des de zero conceptes sobre la IA i Machine Learning i integrar un model funcional en una pàgina web.
- **O4: Utilitzar bones pràctiques de programació en el disseny del sistema.** Investigar sobre bones pràctiques a l'hora de construir i dissenyar una pàgina web, fent-la segura i efectiva.

1.5.2. Possibles obstacles i riscos

Durant el desenvolupament del projecte poden aparèixer obstacles i riscos que poden provocar aturades o fer que el projecte no sigui complet [1]. Els obstacles i riscos observats són els següents:

Potència computacional. A l'utilitzar un model de Machine Learning, perquè funcioni de forma correcta s'ha d'entrenar i passar una sèrie de filtres que requereixen una potència computacional molt gran i poden durar hores. És per això que a l'hora de debugar es poden invertir una quantitat de temps inesperada i que pot afectar en el desenvolupament del projecte.

Falta d'experiència en noves tecnologies. Per construir una pàgina web es necessiten coneixements *Full-stack* i freqüentment hi treballen diverses persones en aquest tipus de projectes. A l'haver un únic desenvolupador, aquest haurà de tenir diferents rols i aprendre diferents tecnologies que poden endarrerir el procés de desenvolupament.

Data d'entrega del projecte. La data d'entrega està fixa i no és variable. Aquest fet pot causar una pressió addicional i, fins i tot, fer una entrega incompleta de la pàgina web causant el no compliment dels objectius descrits anteriorment.

Organització de tasques. Abans de començar el projecte s'ha de fer una organització de les tasques que s'han de dur a terme. Si aquesta organització és errònia, és molt optimista o no s'aprofita el temps completament, pot derivar a problemes de disseny i, per tant, no lliurar la solució escollida.

1.6. Intel·ligència Artificial

L'habilitat de decidir si una frase és positiva, negativa o neutra és una de capacitat de l'ésser humà que, mitjançant un raonament, dona un resultat, majoritàriament, encertat. La intel·ligència artificial ens permet, exactament, portar aquest comportament en una màquina. [38]

Avui dia, múltiples sistemes es recolzen en la IA per tal d'imitar la intel·ligència humana en realitzar diferents tasques que no són arbitràries. Per exemple, els xats de bots es fan ressò d'aquesta tecnologia per donar respostes a unes preguntes de forma realista i que l'usuari no tingui la sensació de parlar amb una màquina programada per tal d'enviar uns missatges determinats quan es rep una entrada qualsevol.

Aquesta idea, encara que pot sonar molt moderna, es remunta al 384 aC, on Aristòtil va ser el primer a descriure un conjunt de regles que representarien una part del funcionament de la ment humana i els seus raonaments per obtenir conclusions racionals. [39]

Alan Turing va proposar el test de Turing en el qual una persona havia d'avaluar una conversa entre un humà i una màquina, dissenyada per donar respostes el més humanament similars, i decidir quins dels dos és l'humà i quin és la màquina. Sorprenentment, s'ha demostrat que *Alexa*, la IA d'Amazon, no és capaç de passar el test de Turing, encara que ja fa setanta-dos anys d'ençà que Alan Turing el va introduir. [40]

Com es pot observar, la idea d'una màquina capaç d'acomplir tasques que són únicament resolubles per un humà fa molt de temps que hi és amb nosaltres, encara que la tecnologia hi és present des de fa relativament poc.

Però, si la IA cerca imitar el comportament de la ment humana, hauria de ser capaç també d'aprendre com ho fem nosaltres o no és capaç? És en aquest aspecte quan s'introdueix l'aprenentatge automàtic o *Machine Learning*. [41]

1.6.1. Machine Learning

El *Machine Learning* és una sub branca de la intel·ligència artificial que va més enllà i té com a objectiu aprendre a partir d'unes dades i unes prediccions. Així mateix, com el pensament lògic humà sap que un plàtan és groc i si està madur, a partir de l'experiència i extreure aquesta informació, una màquina hauria d'arribar a la mateixa conclusió si li proporcionem les dades necessàries i sense definir aquest fet explícitament. [42]

Aquesta és la principal premissa que diferencia un sistema de IA amb un sistema de ML. El sistema de IA ha de ser programat per tal de donar unes respostes similars al pensament humà i el sistema de ML no ha de ser programat, tan sols ha de ser entrenat amb les dades adients.

1.6.2. Recurrent Neural Network

La part més important d'un model de *Machine Learning* és la xarxa neuronal. Aquesta xarxa neuronal és la que s'encarrega del processament de dades en un entrenament i finalment de les prediccions del model. Hi ha diferents tipus de xarxes neuronals cadascuna d'elles orientada a un objectiu diferent. Per exemple les CNN, xarxes neuronals convolucionals, s'utilitzen pel reconeixement d'imatges, ja que utilitza blocs per a mapejar diferents parts d'una imatge. [46]

Encara que hi existeixen moltíssims tipus de xarxes neuronals, i cada vegada són més interessants, jo faré servir una xarxa neuronal recursiva o RNN. Aquesta classe de xarxa neuronal és ideal per al processament de text perquè empra el temps. És a dir, altres xarxes neuronals reben una sèrie d'entrades de dades i la processen cap a la següent capa. En canvi, les RNN processen una sèrie d'entrades de dades i no només la processen cap a la següent capa sinó també cap a la següent neurona de la mateixa capa. Això és molt adequat per a tipus de prediccions que involucren dades que segueixen una seqüència, ja que es recorda i es processa informació sobre l'estat actual i l'estat en un instant passat per recordar informació que ja ha sigut processada, cosa que ens permet donar context. [44]

Per exemple, tenim la frase "*He gaudit de les vacances encara que no hagi viatjat*". En aquesta frase es pot observar com separant la frase en dues oracions: "*He gaudit les vacances*" és positiva i "*encara que no hagi viatjat*" és negativa (si t'agrada viatjar). Si utilitzem una xarxa neuronal sense recursivitat donaria un resultat negatiu, ja que processa l'última part de la frase que té una connotació negativa. En canvi, amb una RNN la positivitat de la primera part de la frase ha sigut processada i portada fins al final de la frase per tal de donar context a aquesta última part i decidir, exitosament, que la frase és positiva.

Però la memòria és finita i si processem una frase més llarga, probablement, el processament de la primera part de la frase no arribi a l'última part de la frase i doni un resultat equivocat. És per això que existeix una subclasse de les RNN anomenada Long-Short Term Memory o LSTM.

1.6.3. Long-Short Term Memory

Hi ha diferents tipus de subclasses d'una RNN, totes elles tenen una millora que les diferencia o estan orientades a un tipus d'objectiu molt específic. Entre aquestes subclasses hi existeix la Long-Short Term Memory o LSTM que és la que s'utilitzarà en aquest sistema.

[45]

Com he dit anteriorment, un dels desavantatges de les Recurrent Neural Networks és la dependència a llarg termini entre els nodes d'aquesta. Les LSTM s'encarreguen d'arreglar aquest error, fent que la informació que es comparteix entre els nodes no canviï i, per tant, arribar a l'últim node de forma inalterada.

Aquestes xarxes neuronals són molt usades en el reconeixement d'escriptura, reconeixement de veu i, fins i tot, en sistemes per trobar anomalies en el tràfic de xarxa, ja que aporten una precisió major en aquelles entrades que les separa un interval de temps major l'habitual.

Per exemple, en 2015, Google va començar a fer ús de LSTMs en els sistemes de reconeixement de veu de Google Voice i estudis recents mostren que Facebook realitza al voltant de 4,5 bilions de traduccions automàtiques cada dia utilitzant LSTMs. **[20]**

Entre altres RNN es poden trobar les GRU, Gated Recurrent Network, que utilitzen el mateix principi de les LSTMs, però ofereixen una simplificació que afavoreixen el rendiment i decreix la precisió. Sent, aquesta, una tecnologia tan recent és molt possible que durant els pròxims anys, o fins i tot mesos, aparegui una nova subclasse de RNNs que millori les LSTMs i introdueixi nous conceptes.

1.7. Estructura del document

Aquest document s'estructura de la següent manera. En el **Capítol 1** s'explica el propòsit i objectius d'aquest projecte i una explicació sobre la intel·ligència artificial i els models de *Machine Learning* que s'utilitzaran per a que el lector/a tingui una idea del que es vol construir. En el **Capítol 2** es fa una anàlisi de mercat on s'analitzen les diverses solucions alternatives que ja existeixen actualment i que les diferencia amb la solució escollida per a aquest treball. En el **Capítol 3** es detalla la metodologia i recursos que es fan servir en el desenvolupament de l'aplicació. En el **Capítol 4** es fa una planificació inicial de les tasques del projecte. En el **Capítol 5** es mostra un diagrama de Gantt d'aquestes tasques. En el **Capítol 6** es fa un informe sobre la gestió de riscos i obstacles que poden aparèixer durant el transcurs del projecte. En el **Capítol 7** es detalla un pressupost del projecte. En el **Capítol 8** s'identifiquen els requisits funcionals i no funcionals del projecte. En el **Capítol 9** es realitza una especificació del sistema, fent ús d'esquemes UML i diagrames de casos d'ús. En el **Capítol 10** es defineix el disseny de l'arquitectura del sistema i els seus components. En el **Capítol 11** s'explica la implementació del sistema i els recursos que s'utilitzen en cada component. En el **Capítol 12** es realitza una validació dels requisits definits en el capítol 8. En el **Capítol 13** es realitza un estudi de bones pràctiques d'enginyeria en sistemes *software* i components de *Machine Learning*. En el **Capítol 14** és dur a terme la justificació de competències transversals especificades al començament del projecte, un informe de sostenibilitat i els resultats de la planificació realitzada en el capítol 4. En el **Capítol 15** es realitzen les conclusions del treball, és a dir, els objectius complerts, possible treball futur i una valoració personal. Finalment, hi existeix una **Bibliografia** amb les referències utilitzades i un **Annex** amb un manual d'usuari per fer servir l'aplicació.

2. Justificació

Durant aquest capítol s'explicarà l'estudi de mercat que s'ha realitzat abans del desenvolupament de l'aplicació, juntament amb una descripció de cadascuna de les alternatives que existeixen ara mateix i que implementen funcionalitats molt interessants. Finalment es detallarà la solució escollida per al desenvolupament d'aquest projecte, fent referència a l'estudi realitzat.

2.1. Solucions alternatives

Abans d'escollir els objectius del Treball de Fi de Grau i el tema principal, s'ha fet un estudi de mercat veient que solucions alternatives existeixen, què manquen i com es poden millorar. A continuació s'explicaran les alternatives que hi ha disponibles:

Hashtagify. Hashtagify és una pàgina web on es poden traquejar hashtags tant de Twitter com d'Instagram. Aquesta pàgina web ofereix una gran quantitat de serveis, tant poder veure els països on la recepció del hashtag ha sigut major com gràfiques que mostren l'evolució d'aquest durant el temps i que ajuden a donar una visualització intuïtiva [5].

L'únic problema d'aquesta pàgina web és que algun dels seus serveis més importants són de pagament i no ofereix una anàlisi de sentiment per poder veure les reaccions emocionals dels usuaris.

Talkwalker. Talkwalker és una aplicació d'escriptori que analitza contingut online tant de xarxes socials com de televisió i ràdio [4]. Aquesta aplicació és capaç de monitorar hashtags a temps real i obtenir una resposta immediata, a més de disposar d'un sistema de reconeixement d'imatges per analitzar imatges de xarxes socials.

Una de les funcionalitats més rellevants d'aquesta aplicació és la generació automàtica d'informes analitzant diferents activitats. Igual que l'anterior cas aquesta aplicació és completament de pagament i no ofereix cap servei de forma gratuïta passat el període de prova.

Brand24. Brand24 és una empresa que disposa d'una aplicació d'escriptori com també Talkwalker [3]. Aquesta aplicació és l'única de les 3 que ofereix el servei d'anàlisi de sentiment, és per això que és la més completa. Ofereix un servei de notificacions que dona a l'usuari una experiència més personal.

És necessari mencionar que aquest software no es centra en anàlisi de hashtags sinó de mencions online en diferents xarxes socials. El problema d'aquesta aplicació és el mateix que el de l'anterior. Tots els seus serveis són de pagament, sense oferir cap servei de forma gratuïta. A més, el públic objectiu són empreses i no usuaris amb curiositat o que busquen analitzar hashtags per compte propi.

2.2. Solució escollida

La solució escollida ha sigut producte de l'estudi prèviament fet d'on s'han aprofitat idees ja implementades en solucions alternatives. Gràcies a aquest estudi de mercat s'han pogut observar mancances en aquestes solucions, directament proporcionant noves idees per implementar i donar a aquest projecte una visió diferent de les alternatives i que cobreixi aquestes mancances i errors que es poden observar.

La solució consisteix en una pàgina web interactiva amb l'usuari, el qual introduirà un conjunt de paraules o un hashtag de la seva elecció i la web li proporcionarà tota la informació que hem catalogat com imprescindible. La informació més important i que forma part del *core* de l'aplicació és el *sentiment analysis*, cosa que, com hem pogut considerar en l'apartat anterior, no hi és present en cap de les solucions descrites.

A més a més s'oferirà informació sobre els missatges amb més reaccions, tants comentaris com *likes*, juntament amb un mapa on es pot veure d'on venen la majoria d'interaccions i poder fer un informe més complet. L'anàlisi mencionada anteriorment serà interactiu, amb capacitat de poder observar que *tweets* són positius, negatius i neutrals. A més, un usuari podrà notificar al sistema si una predicció és errònia per tal de poder reentrenar el model.

No només això, sinó que l'aplicació tindrà gestió d'usuaris, és a dir, podràs loguejar-te en el sistema per poder guardar les anàlisis que hakis realitzat i poder observar-los posteriorment. Gràcies a això, existirà la possibilitat d'afegir amics que podran visualitzar el teu perfil i, per tant, poder analitzar que anàlisis t'has guardat i els resultats d'aquests.

Per terminar, també hi haurà una secció on poder veure quines són les cerques més populars a Twitter i on l'usuari actiu pot investigar encara que no tingui cap cerca en ment. L'usuari podrà filtrar aquesta secció per país i temps per tal de tenir una experiència més personal i engrandir la mostra de resultats.

A continuació es presenta la **Taula 2** on es pot veure d'una forma més visual les explicacions donades en els apartats anteriors on es comparen les altres solucions a la nostra. En aquesta taula es pot observar quines solucions disposen de les característiques esmentades o funcionalitats més rellevants. Dit això podem treure en conclusió que la nostra solució és molt superior pel que fa a funcionalitats claus per a l'usuari i serveis disponibles.

Característiques	Hashtagify	Talkwalker	Brand24	Nostra solució
Pàgina web	Si	No	No	Si
Aplicació escriptori	No	Si	Si	No
Sentiment Analysis	No	No	No	Si
Servei gratuït complet	No	No	No	Si
Compartir anàlisis	No	No	No	Si
Perfil d'usuari	No	Si	Si	Si
Top hashtags	Si	No	No	Si
Generació de informes	No	Si	No	Si
Anàlisi de hashtags	Si	Si	No	Si
Open Source	No	No	No	Si

Taula 2: Comparativa entre solucions

3. Metodologia i rigor

En aquest apartat del document explicaré que metodologia utilitzaré durant la realització del projecte i a través de què elements m'ajudaré per tal de facilitar la gestió i assegurar el bon desenvolupament d'aquest.

3.1. Mètodes de treball

Per a la realització del projecte he decidit utilitzar una metodologia *Agile* com és SCRUM. Aquesta metodologia ens ajudarà a dividir el projecte en diferents *Sprints* on cadascun d'aquests conté tasques a completar en un període de temps determinat [8]. Al final de cada *sprint* es farà un *sprint review* on es presentarà els avenços al director i codirectora del projecte per així assegurar que el que s'està construint és de bona qualitat i poder realitzar canvis importants a temps sense alterar la realització del projecte [6]. No només això sinó que el nostre projecte també ha de complir amb els mètodes de treball de sistemes basats en IA que ens indiquen com desenvolupar un projecte ajuntant intel·ligència artificial i DevOps [27] i que detallarem en l'apartat 3.1.4. [29]

3.1.1. Sprint planning

El Sprint planning consisteix en una reunió entre els integrants de l'equip per decidir quines tasques es realitzaran durant el Sprint corresponent, que sol durar unes dues setmanes, depenent el projecte. A més a més, també es decideix a qui estaran assignades les diferents tasques que s'ha decidit incloure en el Sprint. En el meu cas, en ser l'únic integrant, s'ha de fer prèviament una estimació de Sprints al començament del projecte on l'objectiu ha sigut equilibrar la càrrega de treball entre Sprints tenint en compte que només hi ha un sol desenvolupador. Aquest Sprint planning s'ha pogut observar amb anterioritat en la planificació inicial del treball.

3.1.2. Sprint review

El Sprint review consisteix en una reunió entre els desenvolupadors d'un projecte i els seus *stakeholders*. Aquesta reunió és una de les més importants, ja que és la que dona forma a la metodologia *Agile* i permet aconseguir un dels seus objectius que és la constant comunicació entre desenvolupadors i *stakeholders* que augmenta la taxa d'èxit d'un treball. Aquesta reunió es produeix al final d'un Sprint i es mostra als *stakeholders* els avenços del projecte i el que està planificat per fer el següent Sprint, d'aquesta forma les parts interessades poden fer-se una idea de com va el desenvolupament del treball i poder fer els

canvis que creguin pertinents en un moment adequat sense alterar el mapa de treball del projecte. En el meu cas, aquestes reunions es faran amb el director i codirectora del treball que actuaran com *stakeholders*.

3.1.3. Sprint retrospective

Finalment, el Sprint retrospective ens serveix per posar en comú que ha anat bé i que ha anat malament en un Sprint. Els integrants de l'equip es reuneixen i donen la seva opinió en què es pot millorar per al següent Sprint. Aquestes reunions són molt útils, ja que permeten organitzar el Sprint planning següent d'una forma més precisa agafant el *feedback* de Sprint retrospective. En el meu cas, després de cada Sprint review es revisarà com ha anat el Sprint review i, juntament amb els comentaris del director i codirectora, es milloraran els següents Sprints.

3.1.4 Scrum i Intel·ligència Artificial

El factor d'integrar intel·ligència artificial en el nostre sistema no només afecta la implementació del projecte sinó també en la forma en el que es desenvolupa. Sabem que Scrum segueix un paradigma de Develop, Test and Deploy, però a l'hora d'afegir un nou component com és la IA aquest paradigma varia.

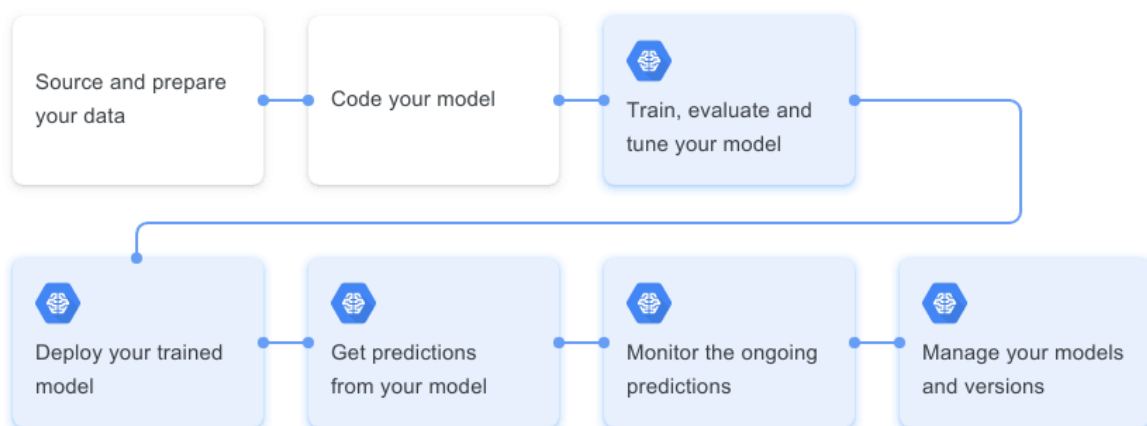


Figura 1: Flux de treball de sistemes basats en IA. Font: Google Cloud

Com podem observar en la **Figura 1**, el flux de treball comença amb els requisits del sistema de IA, una vegada sabem què volem construir i com ho volem fer toca fer una recollida de dades per tal d'entrenar el model. Aquesta recollida no sempre arribarà en un format adient i és per això que hem de "netejar-la".

Una vegada tenim la nostra font de dades, toca desenvolupar el model i testear-lo fins a arribar a una precisió adequada. Aquest pas és recursiu, ja que probablement haurem de fer diferents iteracions canviant la configuració del model per tal d'aconseguir l'objectiu de precisió. Quan hem assolit la precisió que volem podem desplegar el nostre model perquè el puguem fer servir en qualsevol aplicació.

Però no acaba amb el desplegament, una part molt important després de fer el desplegament, és el monitoratge dels resultats i observar que aquests són correctes i en cas de no ser correctes tornar a desplegar el model amb els canvis adients. A continuació en la **Figura 2** es pot veure amb detall l'esquema i les parts de l'aplicació, començant amb la recollida de dades que s'utilitzarà en el desenvolupament del model ML, continuant amb el desenvolupament de l'aplicació que integrarà aquest component ML i, finalment l'estudi de bones pràctiques que s'utilitzaran en el transcurs del projecte.

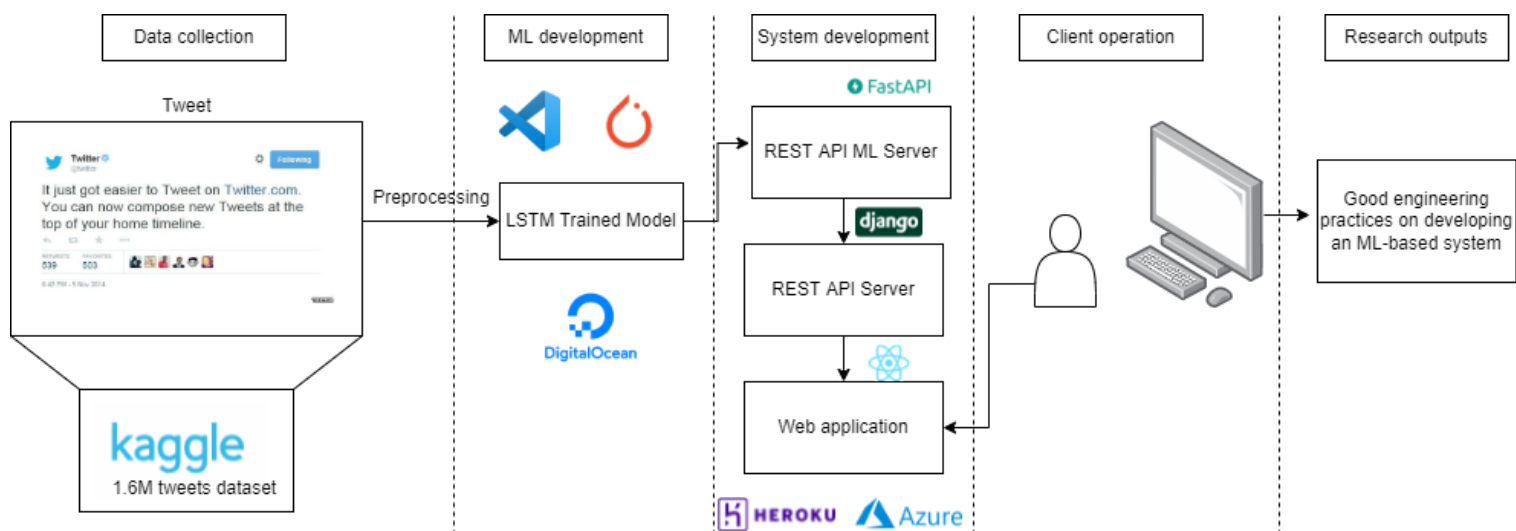


Figura 2: Esquema general de l'aplicació

3.2. Eines de seguiment

Durant aquest subcapítol s'explicaran les diferents eines que es faran servir per assegurar la utilització de la metodologia *Agile* i que m'ajudaran a facilitar el desenvolupament del projecte, sigui la construcció d'aquesta memòria com la implementació del codi font.

3.2.1. Trello

Per poder integrar SCRUM en el nostre projecte ens podem fer servir d'una aplicació de navegador com és Trello. Aquesta aplicació serà de gran utilitat, ja que conté funcionalitats claus per facilitar l'organització de *sprints* i tasques [8]. Amb Trello podem crear un projecte i definir les diferents tasques que es duran a terme durant el transcurs de l'aplicació i dividir-les entre *sprints*. A aquestes tasques podem assignar punts i temps que es pensa que es tardarà a completar. Gràcies a aquestes funcionalitats podem organitzar *sprints* sense una càrrega de treball molt gran o molt petita. A més, ofereix gràfics per poder observar l'evolució de les tasques completades durant un *sprint* i així fer retrospectiva i veure en què es pot millorar pel següent *sprint* [9].

3.2.2. GitHub

GitHub és una eina clau per a qualsevol projecte *software*, no només és útil per a projectes amb més d'un desenvolupador, sinó que també és útil per a implementar integració contínua, guardar el codi en un repositori en cas de perdre localment, distribuir les diferents funcionalitats per branques i tenir una gestió de codi més neta i clara.

S'utilitzarà dues branques, la branca *master* i la branca *develop*, aquestes branques, encara sent només un desenvolupador, ens ajudaran a mantenir una gestió de codi correcta. La branca *develop* s'utilitzarà a l'hora de desenvolupar les funcionalitats que s'especifiquin, una vegada aquestes funcionalitats funcionen correctament, es farà una *pull request* per integrar aquestes funcionalitats a *master* on s'executaran els tests corresponents i es desplegarà l'aplicació.

3.2.3. Google Drive

S'utilitzarà Google Drive per emmagatzemar els documents corresponents al desenvolupament del projecte, ja sigui imatges, documents informatius, etc. També s'utilitzarà Google Drive a l'hora de desenvolupar la memòria del TFG, com també l'informe de seguiment. Aquesta eina ens és molt útil per poder guardar documents en el Cloud i no perdre'ls en cas de desbaratament de l'equip o qualsevol incidència. També es poden compartir documents, cosa que ens resultarà molt útil a l'hora de rebre *feedback* per part dels *stakeholders*.

3.2.4. mllint

mllint és un *linter* de Machine Learning. És una utilitat de línia de comandes per avaluar la qualitat tècnica dels projectes d'aprenentatge automàtic (ML) i intel·ligència artificial (IA) escrits en Python mitjançant l'anàlisi del codi font del projecte, les dades i la configuració d'eines de suport [7]. Aquest *linter* ens ajudarà a crear codi de bona qualitat i així complir un dels objectius proposats anteriorment. A més de testejar el model d'aprenentatge automàtic i ajudar en el desenvolupament i evolució d'aquest.

4. Descripció de les tasques

Seguint la metodologia *Agile* les tasques que es duran a terme durant el desenvolupament d'aquest projecte estan agrupades en diferents iteracions. Cadascuna d'aquestes iteracions portarà assignada una data d'inici, una data de final i una quantitat d'hores estimada per la compleció de la iteració. És important mencionar que aquesta planificació, encara feta amb la precisió més gran possible, és subjecte a canvis. El desenvolupament del projecte començarà el dia 20/09/2021 per motiu de l'inici d'un nou curs, i finalitzarà abans de la defensa del treball a finals de gener.

A continuació es descriuen les diferents iteracions i les tasques que contenen:

4.1. Planificació del projecte

En aquesta iteració es durà a terme la planificació del projecte. Aquesta iteració és vital, ja que tindrà un fort impacte en l'èxit del projecte. El cost en hores d'aquesta iteració és un total de 60.25 hores. El període de temps d'aquesta iteració és del dia 20/09/2021 fins al 18/10/2021.

T1.1. Abast i contextualització. Durant aquesta tasca es contextualitzarà i es definirà l'abast del projecte. S'analitzaran diferents alternatives i es definiran uns objectius, juntament amb la descripció de la solució escollida. A més a més, es documentaran els possibles riscos i obstacles durant el desenvolupament del projecte i la metodologia a seguir. S'estima que aquesta tasca tingui un cost de 24.5 hores.

T1.2. Planificació temporal. Es documentarà una planificació temporal de l'execució total del projecte. Es proporcionarà una descripció detallada de les tasques que es duran a terme, seguint la metodologia descrita prèviament. No només això, sinó que es mostraran estimacions i diagrama de Gantt per definir els requisits entre tasques. Aquesta tasca tindrà un cost de 8.25 hores.

T1.3. Pressupost i sostenibilitat. En aquesta tasca es definirà el pressupost del projecte i un informe sobre la sostenibilitat d'aquest. Es justificaran els costos del projecte, ja sigui el cost per hora dels implicats com els procediments que es duran a terme per controlar aquest pressupost. També es farà una anàlisi detallat de la sostenibilitat del projecte. S'estima que el cost d'aquesta tasca sigui de 9.25 hores.

T1.4. Integració del document final. Finalment, s'agruparan les tasques anteriors en un mateix document i es faran canvis respecte a la qualitat d'aquests. Aquesta tasca és molt important, ja que s'arreglaran les errades en els anteriors documents i s'ajuntarà amb cohesió. Aquesta tasca s'estima que tingui un cost de 18.25 hores.

4.2. Incepció del projecte

En aquesta iteració es durà a terme una anàlisi exhaustiu dels requisits funcionals i no funcionals del projecte. S'estima que aquesta iteració tingui un cost total de 50 hores. La data d'inici d'aquest serà el 19/10/2021 i la data final el 25/10/2021.

T2.1. Definició de funcionalitats. Juntament amb el director i codirectora del projecte es defineixen les diferents funcionalitats del projecte. Ambdós ajudaran al fet que el projecte tingui el nombre adient de funcionalitats, tenint en compte la complexitat i la capacitat de temps. S'estima que aquesta tasca tingui un cost de 10 hores.

T2.2. Definició de requisits de qualitat. Durant aquesta tasca es definiran els requisits no funcionals que necessita el nostre projecte per tenir èxit. Una altra vegada, tant el director com la codirectora, ajudaran en la definició d'aquests requisits per assegurar la qualitat del projecte. S'estima que aquesta tasca tingui un cost de 15 hores.

T2.3. Estudi sobre bones pràctiques. Com s'ha definit en els objectius principals del projecte, s'aplicaran bones pràctiques de programació les quals han de ser identificades mitjançant un estudi de les mateixes i saber quines podem integrar. S'estima que aquesta tasca tingui un cost de 20 hores.

4.3. Primera iteració

La primera iteració del projecte tindrà com a objectiu construir el model de *Machine Learning*, amb una qualitat adient. El període en el qual es durà a terme aquesta iteració serà del 26/10/2021 al 04/11/2021. Tindrà un cost total de 65 hores.

T3.1. Investigació sobre models de *Machine Learning*. Durant aquesta tasca es durà a terme una anàlisi de les diferents formes que existeixen per construir un model

d'aprenentatge automàtic, tenint en compte els propòsits del nostre model i la corba d'aprenentatge. S'estima que aquesta tasca tingui un cost de 10 hores.

T3.2. Construcció del model. En aquesta tasca es construirà el model, tenint en compte les dades obtingudes en la tasca anterior. A més, s'entrenarà el model i es testarà per tenir una precisió adequada. S'estima que aquesta tasca tingui un cost de 20 hores.

T3.3. Integració de mllint al model. Durant el desenvolupament d'aquesta tasca s'integrarà el *linter* mllint per tal de poder assegurar la màxima qualitat del nostre model. S'estima que aquesta tasca tingui un cost de 5 hores.

T3.4. Integració del model amb API de Twitter. En aquesta tasca s'integrarà l'API de Twitter per tal d'extreure els tweets que volem, amb el model de *Machine Learning*. Aquesta tasca ens servirà per analitzar els tweets amb el nostre model. S'estima que aquesta tasca tingui un cost de 30 hores.

4.4. Segona iteració

Durant aquesta iteració es comença a desenvolupar la pàgina web. L'objectiu principal d'aquesta iteració és la de la interacció amb els usuaris. Aquesta iteració tindrà una duració del 05/11/2021 fins al 21/11/2021 i tindrà un cost total en hores de 110.

T4.1. Especificació. Durant aquesta tasca es durà a terme la fase d'especificació del sistema on es construeixen els casos d'ús, a més del model conceptual i les diferents restriccions. S'estima que aquesta tasca tingui un cost de 30 hores.

T4.2. Construcció de la pàgina web. Durant aquesta tasca es farà una anàlisi dels diferents frameworks que existeixen per tal de construir una pàgina web i s'escolliran dos tenint en compte el grau de complexitat i la curva d'aprenentatge. A més es definiran els diferents models que s'utilitzaran, juntament amb les seves propietats. S'estima que aquesta tasca tingui un cost de 30 hores.

T4.3. Log in dels usuaris. Aquesta tasca té com a objectiu implementar un inici de sessió per als usuaris. Es construirà un formulari d'inici de sessió pel qual els usuaris poden accedir al nostre servei. Aquesta tasca tindrà un cost de 25 hores.

T4.4. Registre dels usuaris. Durant aquesta tasca s'implementarà el registre a la pàgina web d'un usuari. Igual que amb l'anterior tasca es construirà un formulari de registre. Aquesta tasca tindrà un cost de 15 hores, ja que s'aprofita el treball fet en la tasca prèvia.

T4.5. Esborrar un usuari. Aquesta tasca té com a finalitat implementar l'esborrament d'un usuari a la pàgina web. S'eliminaran les credencials d'aquest i no podrà tornar a iniciar sessió amb el seu usuari i contrasenya. S'estima que aquesta tasca tingui un cost de 10 hores.

T4.6. Perfil d'un usuari. Durant aquesta tasca es construirà el perfil d'un usuari on podrà observar les seves dades i modificar-les al seu gust. S'estima que aquesta tasca tingui un cost de 20 hores.

4.5. Tercera iteració

Durant aquesta iteració es duran a terme tasques sobre la integració del model de *Machine Learning* a la pàgina web, juntament amb l'anàlisi de *hashtags*. Aquesta iteració tindrà una duració del 22/11/2021 fins al 06/12/2021. El cost total d'aquesta iteració és de 100 hores.

T5.1. Integració del model de *Machine Learning* a la pàgina web. Durant aquesta tasca s'integrarà el model ja fet a la pàgina web. Aquesta tasca tindrà una duració de 10 hores, ja que el model ja ha sigut construït.

T5.2. Generació de l'anàlisi. Aquesta tasca és la principal funcionalitat del nostre projecte. Es busca generar els diferents gràfics i informació a partir de la cerca del *hashtag* i els resultats del model de predicció. S'estima que aquesta tasca tindrà una duració major a les anteriors, d'un total de 70 hores.

T5.3. Guardar i compartir l'anàlisi. L'objectiu d'aquesta tasca és guardar l'anàlisi en el perfil de l'usuari per tal que pugui accedir en qualsevol moment i compartir aquest de diferents formes. S'estima que aquesta tasca tingui una duració de 20 hores.

4.6. Quarta iteració

Aquesta iteració té com a objectiu refinar les tasques fetes en l'anterior iteració. Com només es tracta d'implementar funcionalitats petites, però complexes, aquesta iteració tindrà una duració del 07/12/2021 fins al 21/12/2021. L'estimació de cost total d'aquesta iteració és de 70 hores.

T6.1. Notificació d'error en una predicció. Durant aquesta tasca es desenvoluparà un sistema de notificacions d'errors en la predicció d'un Tweet. S'estima que aquesta tasca tingui un cost de 20 hores.

T6.2. Interacció amb els gràfics produïts. L'objectiu d'aquesta tasca és la interacció amb les gràfiques desplegadas en l'anàlisi. L'usuari ha de poder interactuar amb elles de forma intuïtiva. S'estima que aquesta tasca tingui un cost de 40 hores.

T6.3. Accedir al Tweet original. Aquesta tasca té com a meta connectar la pàgina de Twitter amb el nostre projecte en clicar sobre un tweet. Aquesta tasca s'estima que tingui un cost de 10 hores.

4.7. Quinta iteració

Durant aquesta iteració es construirà la segona pàgina on es poden observar els *hashtags* de moda. Aquesta iteració tindrà una duració del 02/01/2022 fins al 10/01/2022.

T7.1. Mostrar els hashtags trending. L'objectiu d'aquesta tasca és implementar una nova pàgina on es mostraran diferents *hashtags* i es podrà accedir a ells. S'estima que aquesta tasca tingui un cost de 30 hores.

T7.2. Canviar país i interval de temps. Durant aquesta tasca es durà a terme la modificació d'aquesta pàgina mitjançant els atributs de temps i localització. S'estima que aquesta tasca tingui un cost de 10 hores.

T7.3. Millores d'interfície i rendiment. Aquesta tasca té com a finalitat implementar millores visuals i de rendiment. Aquesta tasca tindrà un cost de 60 hores.

4.8. Sexta iteració

Per finalitzar, durant aquesta iteració es prepararà la defensa del projecte. Aquesta iteració tindrà una duració de l'11/01/2022 fins al 24/01/2022. Tindrà un cost de 40 hores.

T8. Preparació de la defensa. Durant aquesta tasca es treballarà en la presentació oral, juntament amb l'exposició dels continguts i funcionalitats que es volen mostrar. S'estima un cost de 40 hores.

4.9. Comunicació i memòria.

És important mencionar que durant el desenvolupament del projecte i en tota iteració es treballarà concurrentment en la memòria del projecte i es portarà una comunicació continuada entre directores i jo mateix.

T9.1. Comunicació entre directores i desenvolupador. Es faran reunions setmanals, en cas que no sigui possible es faran al final de cada iteració, entre Silverio i Cristina amb jo mateix. Això servirà per portar el projecte de forma adequada i arreglar errors que puguin notificar.

T9.2. Construcció de la memòria. Així mateix, amb la tasca anterior, durant el transcurs de les iteracions es construirà concurrentment la memòria, descrivint i definint cada iteració.

4.10. Recursos

4.10.1. Recursos humans

Els recursos humans són les persones responsables de desenvolupar el projecte. Entre molts rols hem d'escollir aquells que siguin més compatibles amb el tipus de projecte a dur a terme. Entre els escollits són: **programador sènior de backend i programador sènior frontend**, aquests programadors seran responsables de la construcció de la pàgina web, com les seves funcionalitats. Ajudant als dos programadors hi haurà un **programador becari** que donarà suport a ambdós programadors en tasques més senzilles.

A més a més, necessitem un **enginyer de sistemes de IA** que sigui responsable de construir un model eficient en les seves prediccions. Per finalitzar, utilitzarem els serveis d'un **dissenyador UX** per tal que la nostra pàgina web sigui tan atractiva com sigui possible i un **Product Manager** que s'encarregarà de definir les funcionalitats i requisits, a més de mantenir una comunicació efectiva amb els *stakeholders*.

4.10.2. Recursos materials.

Entre els recursos materials que utilitzarem per desenvolupar aquest projecte els podem dividir entre recursos *hardware*, com són dos laptops, un per documentar el progrés i un altre amb més potència computacional per entrenar models, i recursos *software* dels quals ens farem servir **Visual Studio i IntelliJ IDEA**, dos IDE de desenvolupament d'aplicacions, **Atenea i Google Drive** per la part de documentació, **Git** per control de versions, **Office 365** per l'organització de documents i creació de stylesheets i, finalment, **Azure** pel desplegament de la pàgina web.

5. Estimacions i diagrama de Gantt

Iteració	Tasca	Codi	Hores	Dependències
Planificació	Abast i contextualització	T1.1	24.5	
Planificació	Planificació temporal	T1.2	8.25	T1.1
Planificació	Pressupost i sostenibilitat	T1.3	9.25	T1.2
Planificació	Integració del document final	T1.4	18.25	T1.1, T1.2, T1.3
Incepció	Definició de les funcionalitats	T2.1	10	
Incepció	Definició requisits de qualitat	T2.2	15	
Incepció	Estudi sobre bones pràctiques	T2.3	20	
Primera iteració	Investigació sobre models	T3.1	10	
Primera iteració	Construcció del model	T3.2	20	T3.1
Primera iteració	Integració de mllint al model	T3.3	5	
Primera iteració	Integració del model amb Twitter	T3.4	30	T3.3
Segona iteració	Construcció de la pàgina web	T4.1	30	
Segona iteració	Log in dels usuaris	T4.2	25	T4.1
Segona iteració	Registre dels usuaris	T4.3	15	
Segona iteració	Esborrar un usuari	T4.4	10	T4.3
Segona iteració	Perfil d'un usuari	T4.5	20	T4.3
Tercera iteració	Integració del model a la web	T5.1	10	
Tercera iteració	Generació del anàlisi	T5.2	70	T5.1
Tercera iteració	Guardar i compartir l'anàlisi	T5.3	20	T5.2
Quarta iteració	Notificació d'error en una predicció	T6.1	20	
Quarta iteració	Interacció amb els gràfics	T6.2	40	
Quarta iteració	Accedir al Tweet original	T6.3	10	
Quinta iteració	Mostrar els hashtags <i>trending</i>	T7.1	30	
Quinta iteració	Canviar país i interval de temps	T7.2	10	T7.1
Quinta iteració	Millores de interfície i rendiment	T7.3	10	T7.3
Sexta iteració	Preparació de la defensa	T8	40	
Comunicació i documentació	Comunicació amb directors	T9.1	20	
Comunicació i documentació	Construcció de la memòria	T9.2	80	

Taula 3: Estimacions de les tasques.

5.2. Diagrama de Gantt

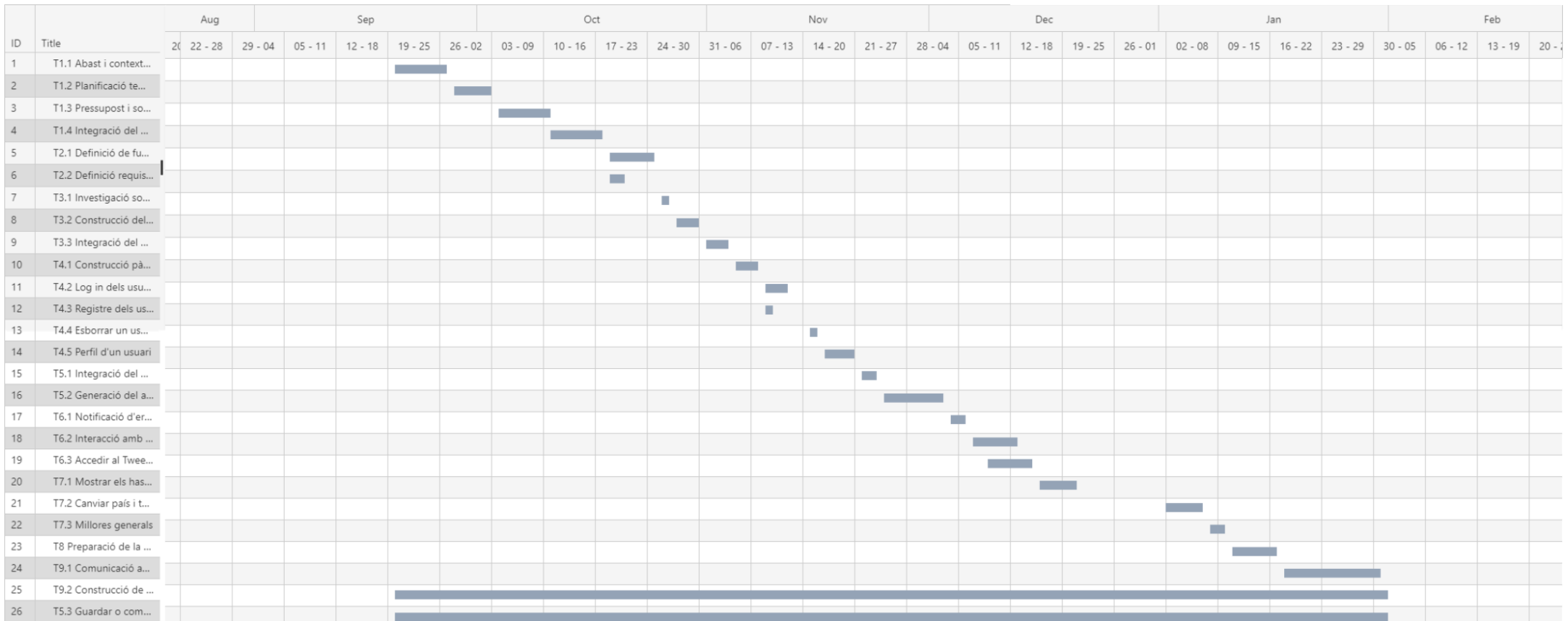


Figura 3: Diagrama de Gantt.

En aquest diagrama de Gantt es mostra les diferents tasques del projecte, com es pot observar algunes tasques es poden fer de forma paral·lel [12]

6. Gestió del risc: Plans alternatius i obstacles

Durant el desenvolupament d'un projecte poden haver-hi moments de bloqueig o obstacles de font externa que alteren el transcurs del treball. És important identificar què pot passar durant el desenvolupament del projecte i tenir un pla de mitigació perquè l'impacte d'aquest problema sigui el menor possible. En anteriors apartats he identificat que dificultats poden aparèixer, ara descriure la forma en la qual hem d'afrontar-los.

Per començar amb la potència computacional és un punt difícil de resoldre, ja que forma part dels recursos que disposem. La possibilitat d'adquirir un nou equip amb més potencia computacional està fora d'abast. Per tant, pel fet que és un problema teòric que no se sap si passarà, en cas que passi disposem dels clouds d'Amazon i Microsoft que disposen de servidors amb potència computacional més gran on podem debugar els models.

Com a segon problema tenim la falta d'experiència en tecnologies que utilitzarem en el desenvolupament d'aquest projecte. Per resoldre aquesta dificultat he fet una planificació comptant amb intervals de temps on només estaré aprenent aquest tipus de tecnologia, com identificant noves tecnologies que ens poden fer millorar el treball.

Continuant amb la data d'entrega final del projecte, el pla de mitigació és aquest mateix document, he fet una planificació del desenvolupament del treball tenint en compte la data d'entrega que hi existeix.

Per últim l'organització de les tasques afecta directament a aquest document, ja que una mala organització és la causa major en treballs de desenvolupament d'aplicacions com aquest. És per això que s'ha tingut en compte petites variacions en intervals de temps de les dates determinades anteriorment per a cada variació, donant un petit marge en cas d'hi hagués contratemps externs i no es poguessin complir les tasques descrites.

Com es pot observar, s'ha pensat plans alternatius per a cada tipus d'obstacle que he identificat. Això no excusa que poden aparèixer nous problemes no identificats en aquesta iteració i que es necessita un pla de mitigació propi.

7. Pressupost

7.1. Identificació dels costos

Durant aquest apartat s'identificaran els diferents costos que es duran a terme per tal que el projecte sigui exitós. Per tenir una organització clara dividirem els costos en 3 subapartats. El primer serà el cost humà, és a dir, aquelles persones amb diferents rols que desenvolupen el sistema. Per continuar tindrem el cost del *software*, les aplicacions que utilitzarem i que ens ajudarà a fer el projecte. Finalment, tindrem el cost del *hardware*, l'equip que adquirirem per emprar aquestes aplicacions.

7.1.1. Cost humà

Els rols que he identificat que ens seran necessaris per construir el sistema seran els típics d'una aplicació web: programador sènior *frontend*, programador sènior *backend*, dissenyador UI/UX, enginyer de sistemes, programador becari i product manager. A continuació es mostrarà una taula on es defineixen els costos de cada rol, respectivament. Hem de recordar que el cost d'un treballador no només engloba el salari, sinó també el pagament de la Seguretat Social de cada treballador [13] [14].

Rol	Salari brut (€/h)	Cost brut (Salari brut * 1.3) (€/h)
Programador <i>frontend</i>	20,5	26,65
Programador <i>backend</i>	22	28,6
Dissenyador UI/UX	14,3	18,6
Enginyer de sistemes IA	28,9	37,57
Programador becari	9	11,7
Product Manager	24,7	32,11

Taula 4: Taula de cost humà.

Ara que sabem quin serà el cost brut per hora de cada treballador, hem d'identificar que tasca fa cada un d'ells i així poder fer una estimació precisa. Per cada tasca es decidirà quin treballador la durà a terme, les hores que es trigarà i, per tant, el cost total de la tasca. Per més claredat: **PSF** (Programador sènior *frontend*), **PSB** (Programador sènior *backend*), **D** (Dissenyador UI/UX), **ES** (Enginyer de sistemes IA), **PB** (Programador becari), **PM** (Product Manager).

ID	Tasca	Hores	PSF	PSB	D	ES	PM	PB	Cost (€)
T1.1	Abast i contextualització	14,5					14,5		465,6
T1.2	Planificació temporal	8,25					8,25		264,9
T1.3	Pressupost i sostenibilitat	9,25					9,25		297
T1.4	Integració document final	18,25					18,25		586
T2.1	Definir funcionalitats	10					10		321,1
T2.2	Definir requisits de qualitat	15					15		481,65
T2.3	Estudi sobre bones pràctiques	20					20		642,2
T3.1	Investigació sobre models	10				10			375,7
T3.2	Construcció del model	20				15		5	608,55
T3.3	Integració de mllint al model	5				5			185,85
T3.3	Integració del model amb Twitter	30				15		15	695,55
T4.1	Construcció pàgina web	30	10	10	10				611
T4.2	Log in dels usuaris	25	5	10				10	536,25
T4.3	Registre dels usuaris	15	5	5				5	334,75
T4.4	Esborrar un usuari	10	5	5					276,25
T4.5	Perfil d'un usuari	20	15	5					542,75
T4.6	Login de Google	10	2					8	146,9
T4.7	Modificar dades personals	20	15	5					542,75
T5.1	Integració del model a la web	10				10			375,7
T5.2	Generació anàlisi	70	20	20				30	1456
T5.3	Guardar i compartir anàlisi	20	5	5				10	393,25
T5.4	Dissenyar trofeus	10			10				186
T5.5	Aconseguir trofeus	15	5	5				5	334,75
T6.1	Notificar error de predicció	20	10			5		5	512,85
T6.2	Interacció amb gràfics produïts	40	20	15				5	1020,5
T6.3	Accedir al Tweet original	10		10					286
T6.4	Recuperar contrasenya	5	1	4					141,05
T7.1	Mostrar hashtags trending	30	5	10	10			5	605,25
T7.2	Canviar país i temps	10	2	3				5	197,6
T7.3	Millores d'interfície	20	5		15				412,25
T9.1	Comunicació (Scrums review)	20					20		642,2

Taula 5: Taula de tasques i cost.

7.1.2. Cost software

Pel cost del *software* identifiquem quins programes utilitzem per desenvolupar el projecte, si són gratis o tenen algun cost addicional i l'amortització que li donarem. En la següent taula es mostraran per a cada *software* el cost i l'amortització, en cas que sigui necessari. Hisenda permet tenir una amortització de 2-3 anys abans de renovar amb actualitzacions, excloent aquell *software* gratuït. Tenint en compte que tindrem una dedicació de 90 dies fins al final del projecte i que el nombre d'hores per dia estimat és de 5. La fórmula per calcular l'amortització del *software* és la següent:

$$\text{Amortització software} = \frac{\text{Cost software}}{\left(3 \text{ anys} \cdot \frac{1100 \text{ hores}}{\text{any}}\right)} \cdot 90 \text{ dies dedicació}$$

Software	Cost	Amortització (€)
Atenea	Gratis	
Google Drive	Gratis	
Office 365 [19]	204.24 €/any	27,85
Git	Gratis	
Visual Studio Code	Gratis	
Intellij IDEA	Gratis	
Microsoft Azure [18]	0,7017 €/hora	38,27
TOTAL		66.12

Taula 6: Taula de cost software.

7.1.3. Cost hardware

Continuant amb els costos d'equip, tenim el *hardware*. Durant el desenvolupament del projecte s'utilitzaran diferents equips per a dividir tasques entre la comunicació i construcció de documents i l'entrenament de models de Machine Learning que necessiten una potència de computació major a la normal. Hisenda ens permet amortitzar el *hardware* 3-4 anys, més tard és obligatori renovar per obsolescència. Seguint amb els càlculs fets anteriorment tenim que la fórmula d'amortització del *hardware* és:

$$\text{Amortització hardware} = \frac{\text{Cost hardware}}{\left(4 \text{ anys} \cdot \frac{1100 \text{ hores}}{\text{any}}\right)} \cdot 90 \text{ dies dedicació}$$

Hardware	Cost	Amortització
HP Laptop 14s-dq2000ns (Portàtil)	549	89,83
MSI Modern 14 B10RBSW (Portàtil)	869	142,2
TOTAL		232,03

Taula 7: Taula de cost hardware

7.1.4. Cost despeses

Per finalitzar tenim el cost de les despeses generals, com per exemple el lloguer de l'oficina on es durà a terme el desenvolupament del projecte. Encara que aquest projecte es desenvoluparà per una sola persona, simularem aquests costos com hem fet en el cost humà. En aquest apartat s'identificaran costos com l'electricitat gastada, costos d'internet, material d'oficina, entre alguns. En la següent taula es mostraran amb més detall, tenint en compte que Hisenda ens permet tenir una amortització de 20 anys.

Despeses	Cost	Amortització (€)
Electricitat [16]	0,37552 €/kWh	12,54
Lloguer d'oficina	2790 €/mes	684,81
Transport	6,8€/mes	1,66
Internet	55,99 €/mes	171,71
TOTAL		870,72

Taula 8: Taula de cost despeses

L'oficina serà la utilitzada actualment en el marc de pràctiques, ubicada en Plaça Francesc Macià i on està disponible 24 hores al dia, 7 dies a la setmana i amb accés a Internet. A 11,6 km de trajecte podem estimar un cost de 1,70€ a la setmana, és a dir 6,8€ al mes. **[17]**

7.2. Estimació dels costos

Per finalitzar amb l'estimació de costos hem de crear una partida de contingències, és a dir, un percentatge que sol ser del 10 al 20% de cada cost per cobrir possibles obstacles no previstos. Es mostrarà amb més detall en la **Taula 9**.

Tipus de cost	Cost (€)	Contingència (%)	Cost total (€)
Cost humà	14478,15	15	16525,65
Cost <i>software</i>	66,12	15	76,03
Cost <i>hardware</i>	232,03	15	266,83
Cost despeses	870,72	10	957,8
TOTAL			17826,31

Taula 9: Taula de contingències.

A més de les contingències hem de calcular el cost de possibles imprevistos que ja hem identificat anteriorment, juntament amb la probabilitat de què apareguin i el temps que trigaria a solucionar-ho.

Imprevist	Probabilitat (%)	Cost estimat (€)	Cost real (€)
Potència computacional	60	799	479,4
Falta d'experiència	40	672,5	269
Organització de tasques	10	235,12	23,5
Dates d'entrega	10	345,56	34,56
TOTAL			806,46

Taula 10: Taula de cost imprevistos

Per finalitzar es mostra el pressupost total del projecte.

Tipus de cost	Cost (€)
Cost humà	14478,15
Cost <i>software</i>	66,12
Cost <i>hardware</i>	232,03
Cost despeses	870,72
Contingències	2179,29
Imprevistos	806,46
TOTAL PRESSUPOST	18632,77

Taula 11: Taula de pressupost total

7.3. Control de gestió

Ja que hem estimat i identificat els costos que portarà aquest projecte, hem de comprovar que aquest pressupost és correcte i les possibles desviacions que hi poden ocórrer. És per això, que hem d'establir una sèrie de regles o funcions per calcular aquestes desviacions. A continuació es mostraran que desviacions es controlaran i de què forma es farà **[15]**:

Desviació de cost humà = *Cost humà estimat - Cost humà real*

Desviació de cost humà per tasca = *(Hores estimades - Hores reals) * Cost real*

Desviació d'hores per tasca = *Hores de tasca estimades - Hores de tasca reals*

Desviació de cost *software* = *Cost *software* estimat - Cost *software* real*

Desviació de cost *hardware* = *Cost *hardware* estimat - Cost *hardware* real*

Desviació de cost despeses = *Cost despeses estimat - Cost despeses real*

Desviació de cost total = *Cost total estimat - Cost total estimat*

Tots aquests càlculs s'introdueixen en un document Excel per tal de seguir una organització dels valors estimats abans del desenvolupament del projecte i que estan presents en aquest document, i els valors reals que es calcularan al final del projecte.

8. Anàlisi de requisits

Després d'indicar i introduir la proposta de treball, juntament amb la planificació i estimació de costos, podem començar a definir els requisits del projecte. Els requisits d'un projecte ens indiquen que s'ha de construir i especifiquen quins criteris hem d'utilitzar per a dur a terme restriccions i la implementació del sistema, com veurem més endavant aquestes preguntes es responen amb dos tipus de requisits que s'especificaran a continuació amb més detall. Durant aquest apartat explicaré què són els requisits funcionals i no funcionals, què informació ens proporcionen ambdós i per què són una peça clau en el desenvolupament d'un projecte, ja no només de *software*, sinó qualsevol mena de projecte.

8.1. Requisits funcionals

En un projecte de *software* se sol dividir els requeriments en dues grans classes: els requisits funcionals i no funcionals (o de qualitat). Durant aquest apartat es descriuran amb el major detall possible els requisits funcionals d'aquest projecte. Els requisits funcionals són, bàsicament, les funcionalitats de les quals disposa el nostre projecte i que ofereix als usuaris actius del sistema. A continuació es detallen els requisits funcionals del sistema amb una descripció del seu respectiu funcionament.

Gestió d'accés al compte

- **Log in / Log out.** Un usuari pot iniciar sessió en el sistema. En cas que tingui compte s'inicia sessió i es mostren el perfil de l'usuari. També pot abandonar la sessió. En cas que no hi hagi compte existent es mostra un error.
- **Log in amb Google.** Un usuari pot iniciar sessió amb les credencials de Google, seguint bones pràctiques de recopilació de dades. Es disposa d'un botó en el qual un usuari pot fer log in amb el compte de Google que desitgi.
- **Registre.** Un usuari pot registrar-se en el sistema amb email i contrasenya. Es mostra un botó a l'hora d'iniciar sessió on l'usuari pot clicar i introduir les seves credencials. En cas que l'email ja hagi sigut introduït al sistema es retorna un error.
- **Registre amb Google.** Un usuari pot registrar-se en el sistema amb els credencials de Google. Es disposa d'un botó el qual un usuari pot prémer i es fa el enregistrament amb els credencials extrets de Google.

Gestió del compte

- **Esborrar compte.** Un usuari pot esborrar el seu compte i les dades que li pertanyen. Es mostrarà un botó en el perfil de l'usuari on pot clicar i esborrar el seu compte del nostre sistema.
- **Modificar contrasenya.** Un usuari pot modificar la seva contrasenya sempre que hi estigui registrat i loguejat, hi hagi introduït l'anterior contrasenya que desitja canviar i la nova contrasenya tingui els requeriments necessaris.
- **Afegir amic.** Un usuari pot afegir a un altre usuari com amic. S'enviarà una sol·licitud a l'altre usuari que pot acceptar o denegar. En cas d'acceptar seran amics, altrament no.
- **Accedir al perfil.** Un usuari pot accedir al seu perfil on pot visualitzar les seves dades personals, sigui amics o hashtags que s'hagi guardat anteriorment.
- **Accedir a perfil d'un amic.** Un usuari pot visualitzar el perfil d'un altre usuari sempre que siguin amics i tingui permisos per visualitzar-ho. Es mostrarà les dades personals que estan disponibles per ser visualitzades, com les anàlisis elaborades i els amics.

Gestió d'anàlisi de Hashtags

- **Anàlisi de hashtag.** Buscar un hashtag per analitzar. Un usuari pot introduir un hashtag o qualsevol paraula en el cercador i es produirà una anàlisi del hashtag cercat.
- **Interactuar amb anàlisi.** Un usuari pot clicar en els resultats de l'anàlisi i se li mostren aquells tweets que han sigut catalogats amb el resultat seleccionat.
- **Accedir al tweet original.** Un usuari pot clicar en un dels tweets i accedir a la pàgina de Twitter on està el tweet per poder interactuar amb ell, directament des de l'aplicació web de Twitter.

- **Notificar predicció errònia.** Un usuari pot notificar si un tweet ha sigut catalogat erròniament. Es disposa d'un botó al costat de cada predicció per tal d'activar la notificació.
- **Guardar anàlisi de hashtag.** Un usuari pot guardar l'anàlisi resultant mitjançant un botó disponible al costat de l'anàlisi. Aquesta anàlisi està disponible per poder consultar-la després en el seu perfil.
- **Compartir anàlisi de hashtag.** Un usuari pot compartir l'anàlisi resultant mitjançant un botó disponible al costat de l'anàlisi. Una anàlisi es pot compartir a aquells usuaris que siguin amics.
- **Veure *hashtags* populars.** Un usuari pot visualitzar els *hashtags* més populars de diferents països i analitzar-los.

8.2. Requisits no funcionals

A continuació es detallaran els requisits de qualitat necessaris per complir els objectius d'aquest projecte. Entre aquests requisits hi estan els proporcionats per la classificació *Volere* i altres característics i necessaris en un sistema amb *Machine Learning* com és el nostre. Els requisits no funcionals ens descriuen com ha de funcionar el sistema per tal que es compleixen els propòsits preestablerts anteriorment. [2]

8.2.1. Requisits no funcionals d'una aplicació web

Tipus de requisit Volere	10a. Appearance Requirements
Descripció	Aquest requisit detalla com ha de ser l'aparença del producte
Justificació	És important que el producte tingui una bona aparença per tal de produir una bona sensació l'usuari
Criteri de satisfacció	Un grup de 5 usuaris, ha de començar a utilitzar el producte en menys de 5 minuts des del primer contacte amb el mateix.

Tipus de requisit Volere	11a. Ease of Use Requirements
Descripció	Aquest requisit detalla com de fàcil ha de ser pels usuaris del producte utilitzar-lo.
Justificació	És important que l'usuari no tingui cap problema en 'identificar les principals funcionalitats del sistema i que sàpiga utilitzar-les sense necessitat d'un entrenament previ.
Criteri de satisfacció	Una enquesta anònima enviada als usuaris un mes abans del llançament ha de mostrar que el 80% d'usuaris no han tingut cap afer

	en utilitzar el producte.
--	---------------------------

Tipus de requisit Volere	12a. Speed and Latency Requirements
Descripció	Aquest requisit detalla la quantitat de temps que es triga a completar diferents tasques.
Justificació	El sistema ha de ser ràpid a l'hora de completar les diferents funcionalitats que l'usuari vulgui.
Criteri de satisfacció	La resposta del sistema ha de ser menor al cap de 10 segons per evitar interrompre el flux de pensament d'un usuari.

Tipus de requisit Volere	12d. Reliability and Availability Requirements
Descripció	Aquest requisit detalla la fiabilitat i disponibilitat necessària del sistema.
Justificació	El sistema ha d'estar disponible en tot moment per a qualsevol usuari que vulgui utilitzar-lo.
Criteri de satisfacció	El sistema haurà d'estar disponible 24 hores al dia, els 365 dies a l'any.

Tipus de requisit Volere	14c. Adaptability Requirements
Descripció	Aquest requisit detalla com s'ha de comportar el producte en altres sistemes.
Justificació	El producte ha d'estar disponible en qualsevol navegador per tal que tot usuari

	pugui tenir accés.
Criteri de satisfacció	El sistema és accessible en tot navegador web, p.e Google Chrome, Microsoft Edge, Mozilla Firefox, etc.

8.2.2. Requisits no funcionals de components de *Machine Learning*

Tipus de requisit Volere	12c. Precision or Accuracy Requirements
Descripció	Aquest requisit detalla que precisos han de ser els resultats generats pel sistema.
Justificació	És important que el model de <i>Machine Learning</i> tingui una precisió adequada per tal de poder fer les prediccions de forma correcta.
Criteri de satisfacció	El model de <i>Machine Learning</i> ha de tenir una precisió mínima del 75%.

Tipus de requisit	Consistency Requirements
Descripció	Aquest requisit detalla a què nivell de consistència han d'arribar les dades mostrades. [23]
Justificació	Les dades mostrades han de ser consistents en les seves prediccions per tal que l'usuari no tingui dubtes de la seva fiabilitat.
Criteri de satisfacció	Les dades han de ser consistents entre entrenaments del model.

Tipus de requisit	Code Quality Requirements
Descripció	Aquest requisit detalla a què nivell de qualitat han d'arribar el codi de la implementació del sistema
Justificació	Fent referència a les bones pràctiques que s'utilitzaran en aquest projecte, és important complir amb una qualitat de codi adient.
Criteri de satisfacció	El codi ha de seguir els criteris PEP8 referenciant la qualitat de codi del model de <i>Machine Learning</i> i el <i>backend</i> i s'arribarà a un 80% de qualitat.

9. Especificació del sistema

Entre les fases del desenvolupament d'un projecte *software* hi existeix la fase d'especificació, tan important, o més, com la fase d'implementació. Just després de la fase d'iniciació del projecte, tenim una visió del que volem en el nostre sistema, encara que no tenim una visió de com fer-ho. Resoldrem aquest problema en la fase d'especificació on decidirem com funcionen les funcionalitats, qui pot utilitzar-les i en què consistirà el nostre sistema.

9.1. Diagrames de casos d'ús

Per començar utilitzarem diagrames de casos d'ús per dividir les funcionalitats tal com hem fet anteriorment i donar un diagrama més visual de què serveis externs necessitaran aquestes funcionalitats i les relacions entre els mateixos casos d'ús i els actors que les activen.

9.1.1. Gestió d'accés al compte

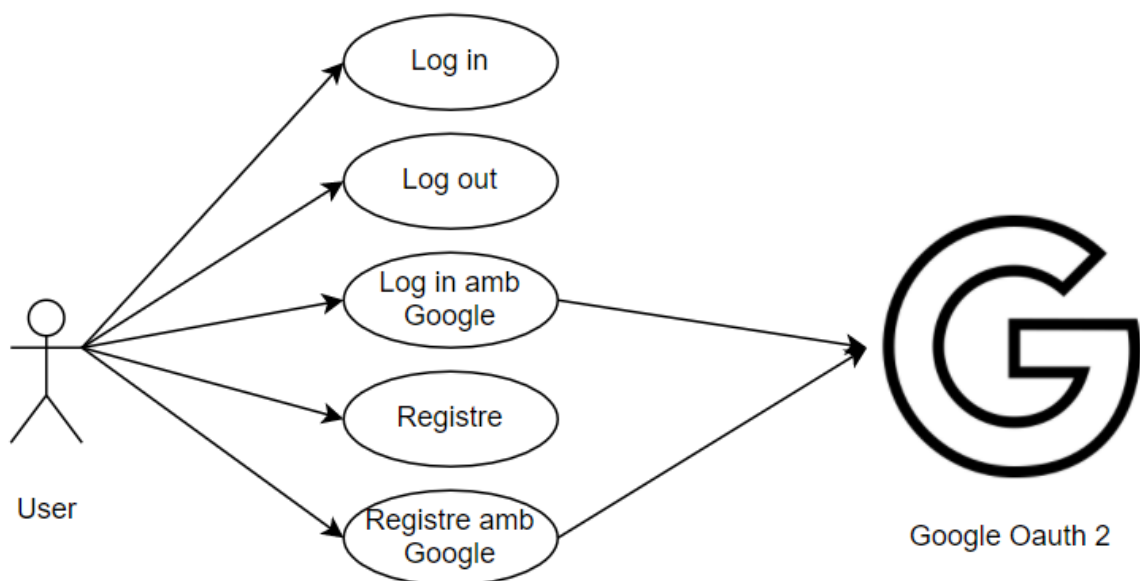


Figura 4: Diagrama casos d'ús de gestió d'accés al compte

9.1.2. Gestió del compte

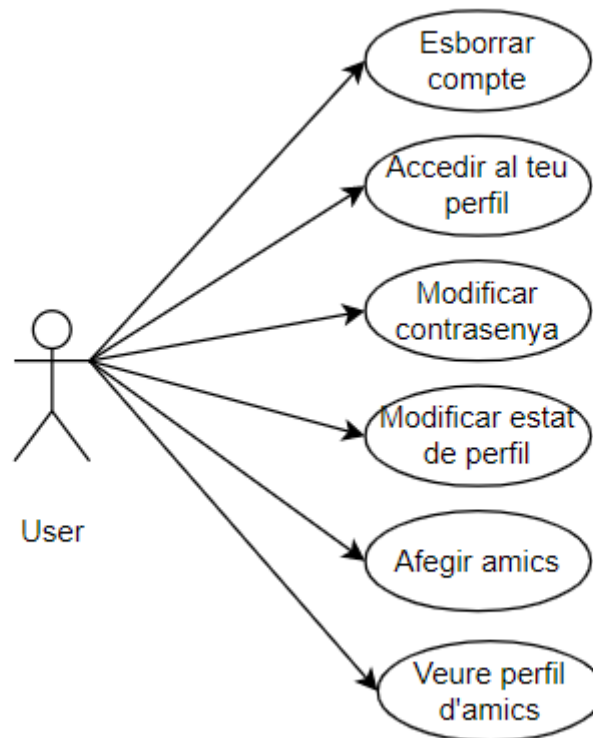


Figura 5: Diagrama casos d'ús de gestió del compte

9.1.3. Gestió d'anàlisi de Hashtags

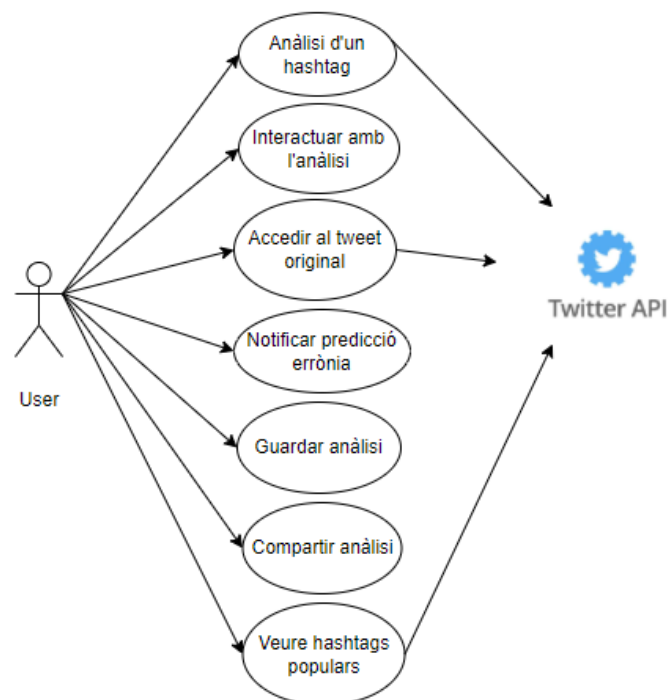


Figura 6: Diagrama casos d'ús de gestió d'anàlisi de hashtags

9.2. Descripció de casos d'ús

9.2.1. Gestió d'accés al compte

- **Log in**

Precondició: L'usuari no està loguejat en el sistema i té compte registrat.

Postcondició: L'usuari està loguejat en el sistema.

Descripció: L'usuari introdueix les seves credencials per tal de loguejar-se en el sistema. Si l'usuari no existeix o la contrasenya és incorrecta el procés dona error.

- **Log out**

Precondició: L'usuari està loguejat en el sistema.

Postcondició: L'usuari no està loguejat en el sistema.

Descripció: L'usuari pot sortir de la pàgina i tancar la seva sessió.

- **Log in amb Google**

Precondició: L'usuari no està loguejat en el sistema i té compte de Google.

Postcondició: L'usuari està loguejat en el sistema amb les credencials de Google.

Descripció: L'usuari pot fer log in amb Google i escollir el compte de Google amb el que vol entrar a l'aplicació.

- **Registre**

Precondició: L'usuari no està registrat en el sistema.

Postcondició: L'usuari està registrat en el sistema.

Descripció: L'usuari introdueix les credencials amb les quals vol identificar-se en el sistema per tal de registrar-se. Si l'usuari ja existeix, el procés dona error.

- **Registre amb Google**

Precondició: L'usuari no està registrat en el sistema i té compte de Google.

Postcondició: L'usuari està registrat en el sistema amb les credencials de Google.

Descripció: L'usuari pot fer el registre amb Google i escollir el compte de Google amb el que vol registrar-se a l'aplicació.

9.2.2. Gestió del compte

- Esborrar el compte

Precondició: L'usuari està loguejat en el sistema.

Postcondició: El compte de l'usuari és eliminat del sistema.

Descripció: L'usuari pot eliminar el seu compte de les dades del sistema. L'usuari no pot tornar a loguejar-se amb els mateixos credencials.

- Modificar contrasenya

Precondició: L'usuari està loguejat en el sistema i la nova contrasenya compleix amb els requisits.

Postcondició: L'usuari està registrat en el sistema amb la nova contrasenya especificada.

Descripció: L'usuari canvia la seva contrasenya seguint els requisits que se li demana i el sistema guarda aquests canvis. En cas que no es compleixi aquests requisits el procés donarà error.

- Afegir amic

Precondició: L'usuari està loguejat en el sistema, existeix l'usuari amic i la sol·licitud és acceptada.

Postcondició: L'usuari es converteix en amic.

Descripció: L'usuari afegeix un usuari com amic, s'envia una sol·licitud i en cas d'acceptada per l'altre usuari es converteixen en amics. En cas de no acceptar-se s'elimina la sol·licitud.

- Accedir al perfil

Precondició: L'usuari està loguejat en el sistema.

Postcondició: S'accedeix al perfil de l'usuari loguejat.

Descripció: L'usuari accedeix al seu perfil per tal de visualitzar les seves dades personals.

- Accedir al perfil d'un amic

Precondició: L'usuari està loguejat en el sistema i l'altre usuari és amic.

Postcondició: S'accedeix al perfil de l'amic.

Descripció: L'usuari accedeix al perfil de l'amic on pot visualitzar les seves dades personals que estan disponibles per ser visualitzades.

9.2.3. Gestió d'anàlisi de Hashtags

- **Anàlisi de hashtag**

Precondició: L'usuari està loguejat en el sistema.

Postcondició: Es genera una anàlisi del hashtag introduït.

Descripció: L'usuari cerca un hashtag i es genera una anàlisi d'aquest.

- **Interactuar amb anàlisi**

Precondició: L'usuari està loguejat en el sistema i s'ha generat una anàlisi.

Postcondició: Es mostren els tweets pertinents a la classificació seleccionada

Descripció: L'usuari pot interactuar amb l'anàlisi per tal de veure resultats diferents.

- **Accedir a tweet original**

Precondició: L'usuari està loguejat en el sistema.

Postcondició: S'obre una pestanya amb Twitter i el tweet seleccionat.

Descripció: L'usuari pot accedir al tweet original en la pàgina de Twitter per poder interactuar amb ell.

- **Notificar predicció errònia**

Precondició: L'usuari està loguejat en el sistema.

Postcondició: Es notifica al sistema que un tweet ha sigut mal classificat.

Descripció: L'usuari pot seleccionar un tweet ja classificat i notificar que la classificació és equivocada.

- **Guardar anàlisi**

Precondició: L'usuari està loguejat en el sistema i s'ha generat una anàlisi.

Postcondició: Es guarda l'anàlisi generada en el perfil de l'usuari.

Descripció: L'usuari pot guardar una anàlisi prèviament generada per poder-hi accedir posteriorment en el seu perfil quan vulgui.

- **Compartir anàlisi**

Precondició: L'usuari està loguejat en el sistema i l'altre usuari és amic.

Postcondició: Es comparteix l'anàlisi amb l'amic.

Descripció: L'usuari comparteix una anàlisi prèviament guardada amb un altre usuari del qual és amic.

9.3. Model conceptual

En aquest apartat definirem el model conceptual de les dades UML, en aquest model conceptual podem observar de forma molt visual com el sistema s'ha de construir, que entitats s'han de tenir en compte i quines relacions han de tenir entre aquestes entitats. Més endavant es detalla cada entitat que representa i els seus atributs.

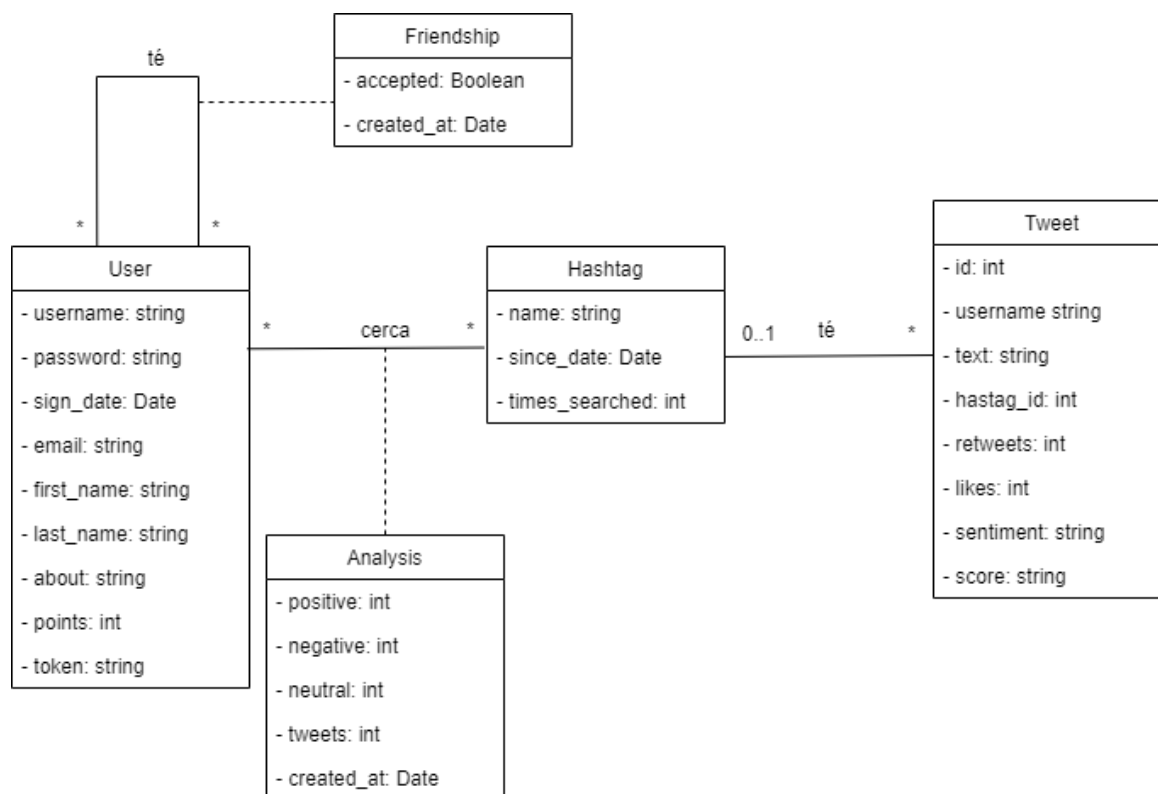


Figura 7: Esquema conceptual UML

User: Representa els usuaris del sistema

- **id:** identificador únic d'un usuari.
- **username:** nom d'usuari únic.
- **password:** contrasenya d'un usuari.
- **sign_date:** data de el primer inici de sessió
- **email:** correu electrònic d'un usuari
- **first_name:** nom real d'un usuari.
- **last_name:** cognom real d'un usuari
- **about:** petita descripció de l'usuari.
- **points:** punts acumulatius d'un usuari.
- **token:** token de l'usuari al registrar-se en l'aplicació

Friendship: Representa una relació d'amistat entre usuaris en el sistema

- **id:** identificador únic d'una amistat.
- **user_id_1:** identificador únic del primer usuari.
- **user_id_2:** identificador únic del segon usuari.
- **date:** data de l'enviament de la sol·licitud d'amistat.
- **accepted:** vertader si la sol·licitud ha sigut acceptada, fals si no.

Analysis: Representa una anàlisi d'un hashtag en el sistema.

- **id:** identificador únic d'una anàlisi
- **user_id:** identificador únic del usuari que genera l'anàlisi.
- **hashtag_id:** identificador únic del hashtag que s'utilitza per l'anàlisi.
- **positive:** nombre de tweets amb sentiment positiu.
- **negative:** nombre de tweets amb sentiment negatiu.
- **neutral:** nombre de tweets amb sentiment neutral.
- **tweets:** número total de tweets analitzats.
- **created_at:** data de la creació de l'anàlisi.

Hashtag: Representa un hashtag en el sistema

- **id:** identificador únic d'un hashtag.
- **name:** nom del hashtag.
- **since_date:** data des d'on es volen agafar tweets per l'anàlisi
- **times_searched:** vegades que s'ha cercat el hashtag.

Tweet: Representa un tweet en el sistema

- **id:** identificador únic d'un tweet
- **tweet_id:** identificador únic d'un tweet en Twitter.
- **text:** contingut del tweet.
- **hashtag_id:** identificador del hashtag que està connectat amb el tweet.
- **retweets:** nombre de retweets d'un tweet.
- **likes:** nombre de likes d'un tweet.
- **sentiment:** etiqueta del sentiment d'un tweet.
- **score:** percentatge del sentiment d'un tweet.

9.3.2. Restriccions d'integritat

- Tweet (Id, PK), Hashtag (Id, PK), Analysis (Id, PK)
- L'username d'un usuari ha de ser únic en el sistema.
- L'email d'un usuari ha de ser únic en el sistema
- La password d'un usuari ha de contenir 8 caràcters com a mínim.
- Un usuari no pot tenir amistat amb ell mateix.
- Un usuari no pot enviar una sol·licitud d'amistat a algú del que ja hi és amic

10. Disseny del sistema

Durant aquest apartat, explicaré el disseny del sistema, una altra peça clau en el desenvolupament d'un projecte. Després d'especificar i definir els requisits podem començar a treballar en el disseny. A continuació es mostrarà l'esquema general del sistema de forma visual perquè sigui el més entenedor possible.

10.1. Arquitectura

En aquest esquema es pot observar una divisió en 6 parts del sistema, la gestió de dades que consisteix a agafar el *dataset* que utilitzarem per entrenar el model, el model de *Machine Learning*, que utilitzarem per realitzar les nostres prediccions, el servidor on es pot accedir al mode, la capa de dades on es gestiona la base de dades, el servidor *backend* on s'implementarà la lògica de l'aplicació i finalment l'aplicació web on l'usuari interactua directament per acomplir les tasques pertinents.

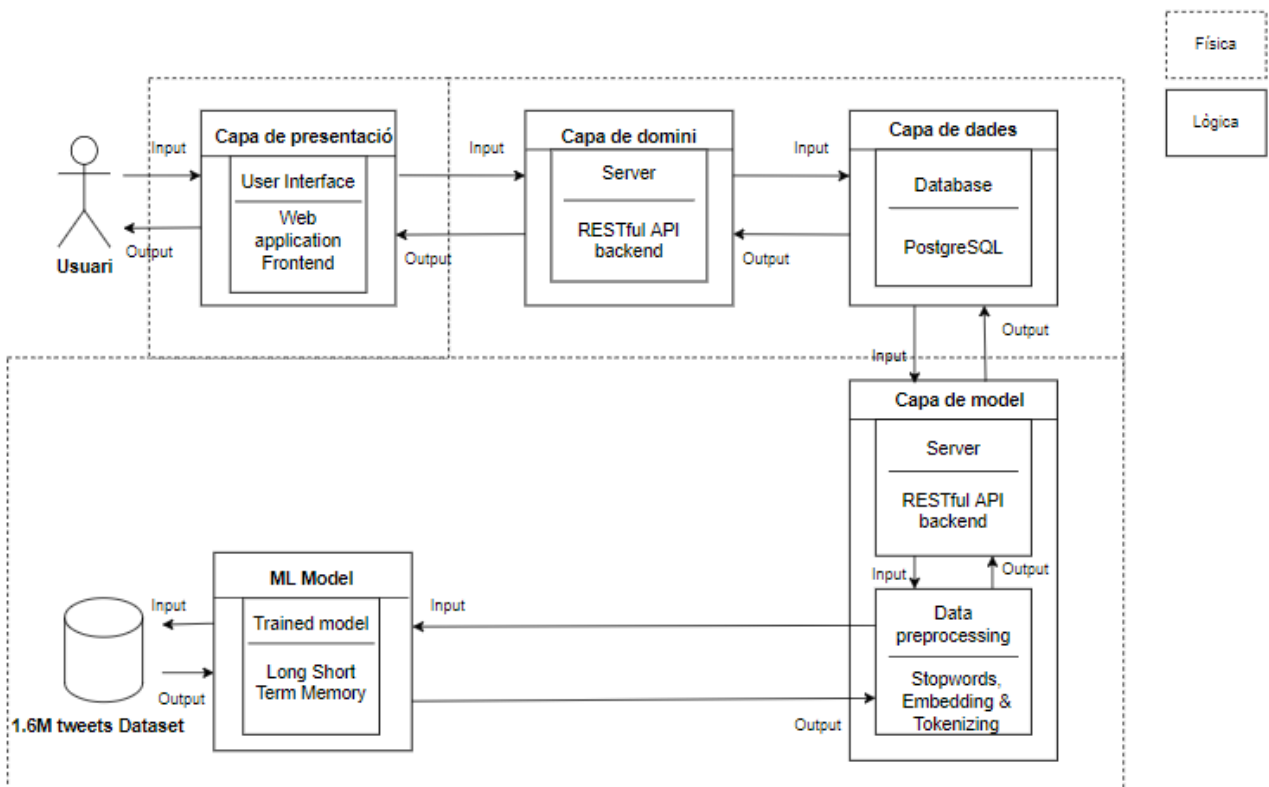


Figura 8: Arquitectura del sistema

10.1.1. Gestió de dades

El nostre disseny de l'arquitectura comença amb el desenvolupament d'un model de *Machine Learning* i per tal d'aconseguir això hem de fer una recollida de dades per tal d'entrenar el model. Kaggle és la major web de *datasets* del món on tothom pot contribuir a millorar els datasets i, fins i tot, construir models i compartir amb els usuaris, és per això que agafem un conjunt de dades existents adients per al nostre sistema.

En aquest conjunt de dades hi existeix un total d'1.600.000 tweets, tots aquests catalogats amb un tag, o etiqueta, que consisteix en un rang del 0 al 4, sent 0 un tweet amb sentiment negatiu, 2 un tweet amb sentiment neutral i 4 un tweet amb sentiment positiu. Hi ha altres camps en aquest *dataset* com, per exemple, l'usuari que ha publicat el tweet, la data de publicació, l'id, però nosaltres només utilitzarem l'etiqueta de sentiment i el text que li correspon.

10.1.2. ML Model

Una vegada hem aconseguit un conjunt de dades fiable i consistent, el podem utilitzar per entrenar el nostre model. Per assolir això faré servir l'IDE Visual Studio Code, el qual és molt útil per poder tenir flexibilitat en aquell que s'està construint i aplicar control de versions. També utilitzarem el servidor de *Jupyter Notebooks* per tal d'entrenar el model, ja que la potència computacional que es necessita és molt gran i això ens ajudarà a agilitzar el procés.

10.1.3. Capa de model

La capa del model de *Machine Learning* consisteix en una petita REST API, amb FAST API, un framework de Python, molt similar a *Flask*, que és perfecte per construir APIs petites que només són accessibles mitjançant una altra REST API com és el nostre cas. A partir de tenir el servidor de ML aquest es desplegarà en DigitalOcean, una plataforma adient per desplegaments de sistemes amb IA, ja que ens permet utilitzar més espai de forma gratuïta que, per exemple, Heroku que només ens deixa utilitzar 500 MB d'espai.

En aquesta capa a més de donar accés al sistema fer ús el model per fer les seves prediccions, també es preprocessa el text per poder ser utilitzat de la forma més precisa possible pel model. Una vegada el text ha sigut processat a una forma entenedora per al model, es genera la predicció.

10.1.4. Capa de dades

La base de dades és un servidor PostgreSQL i és gestionada per *Django*, encara que *Django* es comporta com una capa diferent de l'hora de fer les migracions a la base de dades. Aquesta base de dades és independent de *Django*. És a dir, *Django* dona les eines necessàries per a gestionar aquesta base de dades de forma senzilla sense haver de preocupar-se per clàusules o sentències SQL. Tot i això, la base de dades pot ser gestionada de forma independent, en un cas de necessitat o en cas que es volgués separar servidor i base de dades.

Aquesta base de dades gestiona els usuaris, hashtags cercats, tweets extrets de l'API de Twitter, amistats i anàlisis.

10.1.5. Capa de domini

En la capa de domini es desenvoluparà una REST API més complexa que portarà la lògica de l'aplicació. Aquesta API es construirà amb *Django*, un framework de Python que és perfecte per desplegar aplicacions complexes i ben documentades. El desenvolupament del servidor *backend* es realitzarà amb l'IDE *PyCharm*, aquest IDE és perfecte per desenvolupar servidors en Python, ja que amb la seva instal·lació porta funcionalitats per a qualsevol framework que necessitem, ja sigui *Django*, *Flask*, *Pyramid*, *Web2Py*, etc.

Un cop hem construït la lògica de l'aplicació ens toca desplegar-ho per tal de poder ser accessible més endavant per l'aplicació web. En aquest cas he escollit Heroku per la seva simplicitat, Heroku ens permet implementar CI/CD de forma molt senzilla tan sols connectant el nostre compte i el repositori de GitHub. A més, en separar model i lògica, el volum màxim per desplegar en Heroku es redueix i ens permet desplegar de forma gratuïta sense costos addicionals.

10.1.6. Capa de presentació

Finalment, necessitem una interfície per tal que l'usuari pugui interactuar amb el sistema de forma senzilla i intuïtiva. És per això que es desenvoluparà una aplicació web que farà d'intermediari entre l'usuari i el *backend*. Aquesta aplicació web es desenvoluparà en Javascript, i encara que hi existeixen molts frameworks que són molt compatibles amb les nostres necessitats, hem decidit treballar amb *React*, ja que m'és familiar, intuïtiu de treballar i dinàmic, justament el que estem buscant en la nostra pàgina web.

L'usuari interactuarà amb la pàgina web i aquesta demanarà la informació necessària al servidor *backend* per tal de desplegar-la a l'usuari de forma agradable a la vista. Aquesta aplicació es desenvoluparà amb l'IDE WebStorm, pel fet que he treballat en el passat amb ell i, igual que amb PyCharm, disposa de moltes ajudes per desenvolupar una aplicació web en qualsevol llenguatge i framework.

Aquesta aplicació es desplegarà en *Azure*, ja que encara que el preu és molt elevat en comparació amb altres alternatives, Microsoft ens ofereix una prova gratuïta que nosaltres aprofitarem i a més, la velocitat i les opcions amb Azure fa que la nostra pàgina web sigui molt més eficient que si la despleguem amb Heroku.

10.2. Diagrames de seqüència

Per tal de donar una solució més visual i poder veure com tots els components anteriors interactuen entre ells, durant aquest apartat es mostraran els diferents diagrames de seqüència de l'activitat més important del nostre sistema, des del log in de l'usuari fins a la generació de l'anàlisi.

Per començar un usuari qualsevol, amb un compte ja registrat en el sistema, inicia sessió en l'aplicació, el sistema s'assegura de què aquest usuari estigui registrat en el sistema i retorna una resposta.

Més detalladament, en la capa de presentació, és a dir l'aplicació web, un usuari introdueix les seves credencials, després d'això es realitza una trucada a la capa de domini on el servidor pregunta a la base de dades si aquest usuari existeix, la base de dades dona una resposta a la capa de domini i aquesta produeix la mateixa resposta a la capa de presentació per a donar un resultat visual a l'usuari de l'acció que acaba de dur a terme.

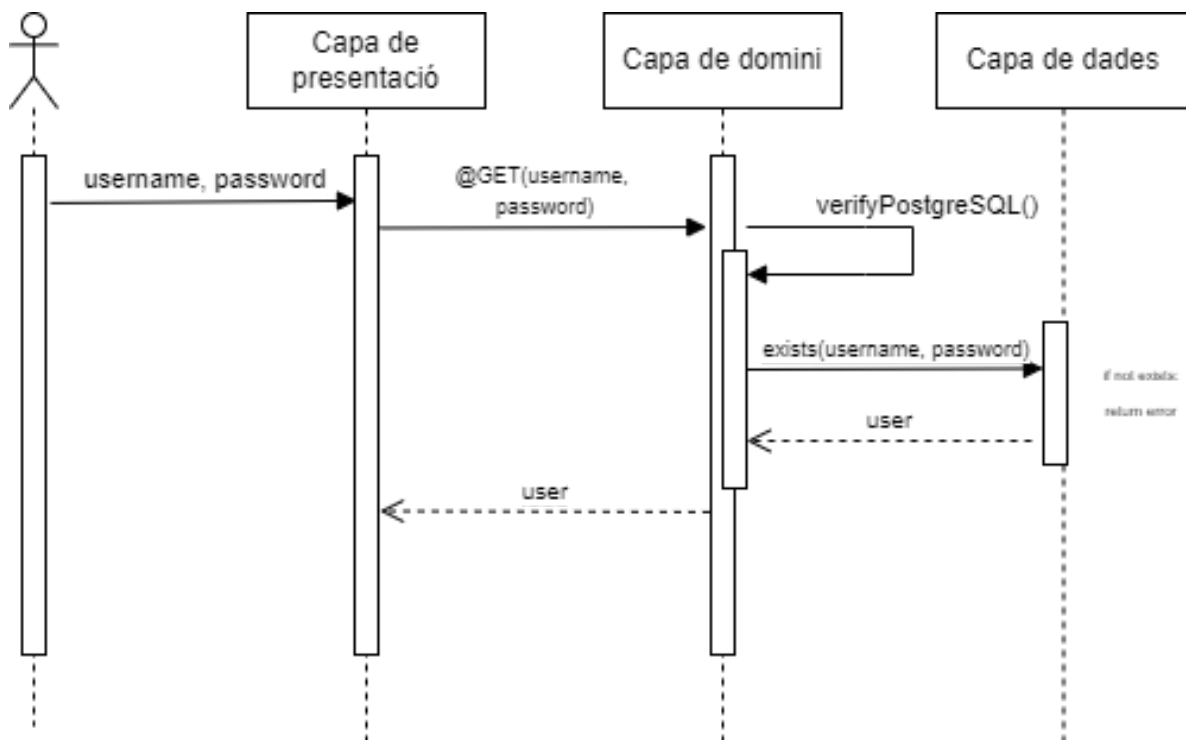


Figura 9: Diagrama de seqüència inici de sessió

Una vegada l'usuari ha iniciat sessió en el sistema decideix cercar un *hashtag* per tal de generar una anàlisi. En la capa de presentació un usuari introdueix el nom d'un hashtag i, opcionalment, una data des de quan agafar tweets per l'anàlisi i el nombre de tweets, per defecte 100. Es fa una crida al servidor de la capa de domini on fa la recoll·lecta de tweets a través l'API de Twitter, cada tweet s'envia al servidor de la capa de model on es fa la predicció del sentiment del tweet i s'insereix en la base de dades.

Una vegada tenim tots els tweets amb la predicció corresponent, es crida a la capa de domini per extreure els tweets de la base de dades que s'han inserit anteriorment. La capa de presentació desplega els tweets, el sentiment respectiu i l'anàlisi corresponent.

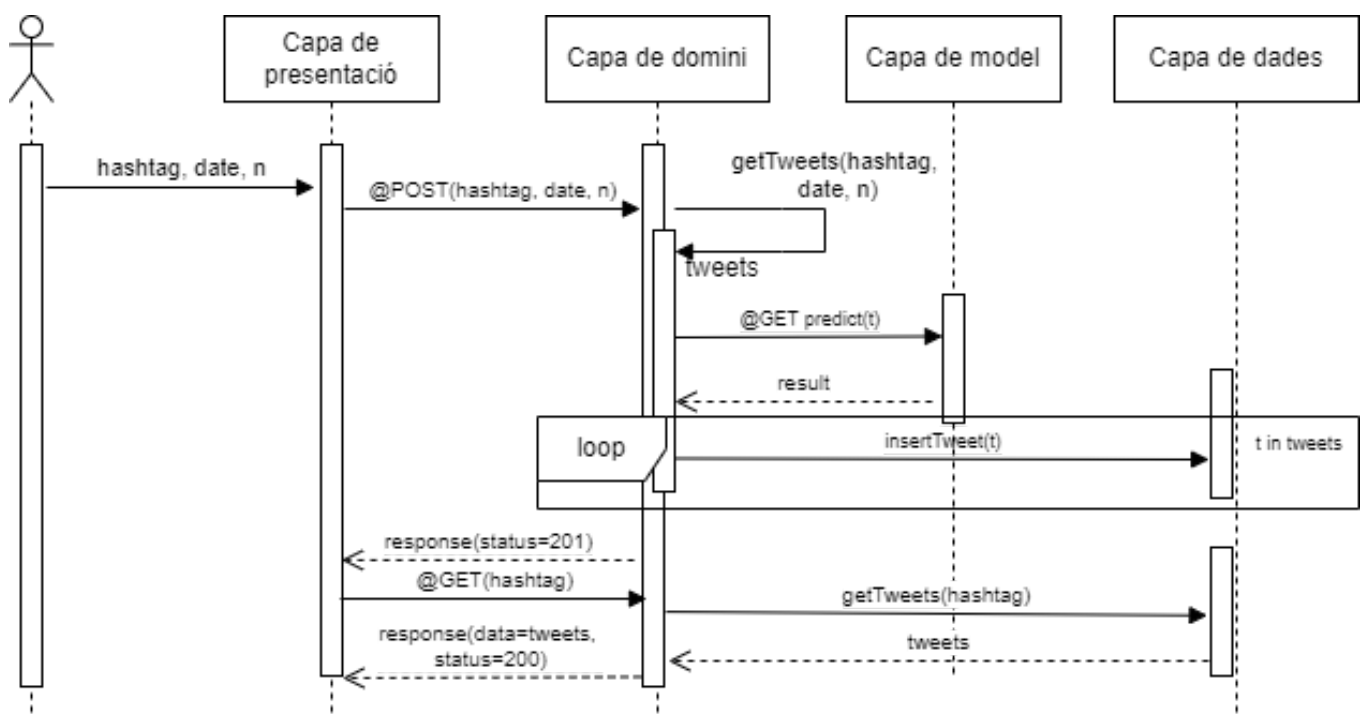


Figura 10: Diagrama de seqüència cerca de hashtag

10.3. Disseny de mockups

Durant l'etapa de GEP es va posar en marxa el disseny de la pantalla principal, la d'anàlisi d'un hashtag, aquests mockups són molt útils per mostrar als *stakeholders* una representació molt visual del producte i introduir millores sense haver de canviar la implementació del sistema.

El mockup que es mostrarà consisteix en la cerca d'un hashtag per l'anàlisi i la mostra de resultats. Com es pot veure en la **Figura 11** inicialment es va dissenyar amb la idea d'ajuntar la cerca amb el mateix anàlisi que apareixia justament a sota del cercador. A més a més, es va pensar a mostrar un mapa del món on es veuria en què país un hashtag ha tingut més influència.

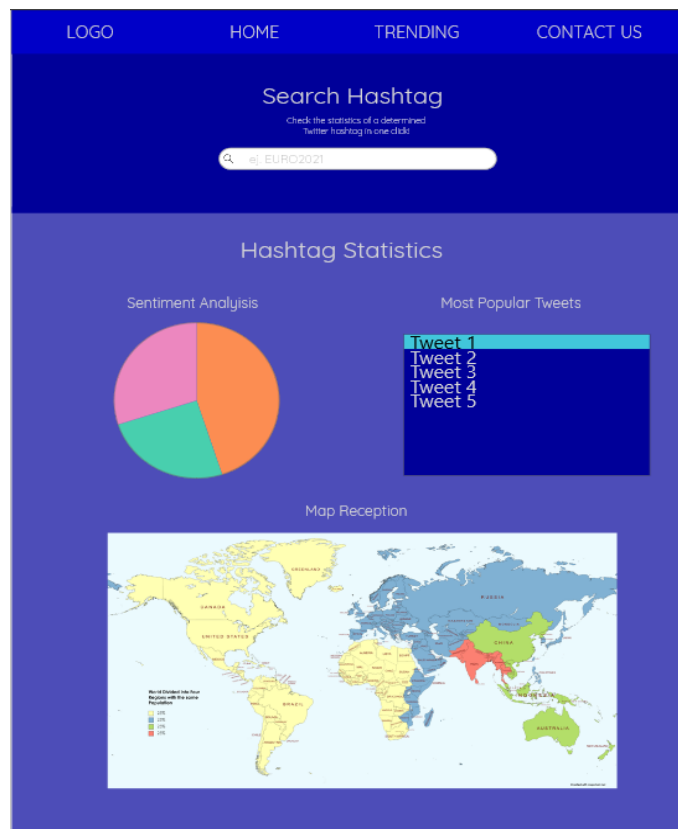


Figura 11: Mockup cercador i anàlisi de hashtags

A continuació es mostra en la **Figura 12** el *mockup* de la pantalla de *trendings*, aquesta pantalla es va pensar inicialment per poder analitzar els hashtags més cercats per als usuaris i poder filtrar aquests resultats per país i data.

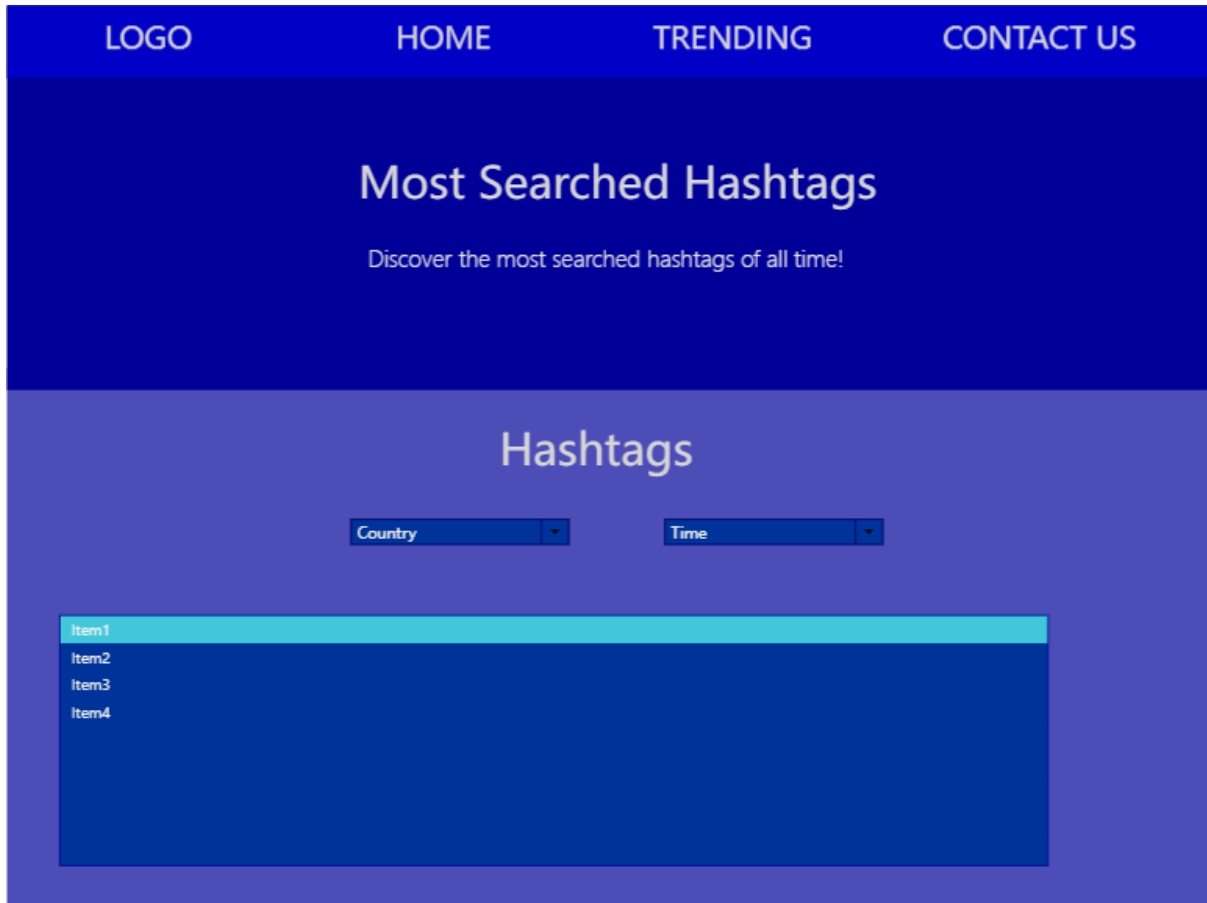


Figura 12: Mockup pantalla de trendings

11. Implementació del sistema

Ara que hem definit un disseny del nostre sistema podem avançar cap al següent pas del desenvolupament *software*, la implementació. Durant aquest apartat s'explicarà, en més detall, els aspectes tècnics de la solució escollida, ja sigui tècniques utilitzades, tecnologies, etc. No només això, sinó que es dedicarà un subapartat per detallar el model de *Machine Learning*, en profunditat.

11.1. Implementació del model

Durant tot aquest document hem estat parlant del model de *Machine Learning* i de com és el centre del nostre sistema. Una vegada hem vist quin és el seu objectiu dins el projecte, veurem com s'ha construït, des de l'elecció del tipus adequat, com els paràmetres específics que ens ofereixen la precisió més gran possible. El model s'ha desenvolupat en l'entorn virtual *Pytorch*, on s'han instal·lat les llibreries de *TensorFlow*, juntament amb *Keras*, que ens faciliten el desenvolupament de models de *Machine Learning* [55]. La implementació del model, com es mostra en la **Figura 13**, [57] comença en el preprocessament de text amb NLP, l'entrenament amb el conjunt de dades escollit i el modelatge per tal de treure la major precisió i eficiència possible.

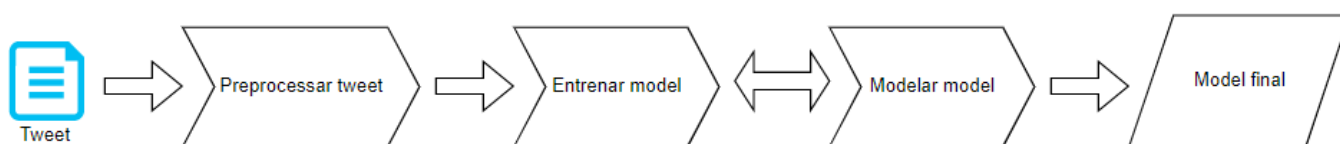


Figura 13: Flow de la implementació d'un model Machine Learning

11.1.1. Natural Language Processing

NLP o Natural Language Processing és un camp de les ciències de computació que s'encarrega en estudiar les interaccions entre ordinadors i el llenguatge humà. Aquest estudi introdueix unes tècniques que s'explicaran a continuació per tal que un ordinador sigui capaç d'entendre el significat d'una oració i, fins i tot, arribar a conclusions que els humans no podem. Aquest conjunt de tècniques ajuden a convertir un text sense cap estructura ni significat per a un ordinador, en unes dades utilitzables d'on es poden treure conclusions [48].

11.1.2.1. Preprocessament de text

Entre aquestes tècniques es troba el preprocessament de dades, aquesta tècnica ens habilita “rentar” les dades i convertir paraules sense sentit per una màquina, en informació important per al model. Aquesta tècnica consisteix en diversos passos per tal de normalitzar una oració i fer que el conjunt de dades sigui consistent. Entre aquests passos es troben: [24]

- **Oracions en minúscula.** Convertir el text en minúscula habilita al model poder predir amb més facilitat i rapidesa.
- **Reemplaçar URLs i noms d’usuari en constants.** Aquestes URLs i noms d’usuari no aporten cap informació important al model, per tant, es poden convertir en constants com: “<url>” o “<user>”.
- **Reemplaçar lletres consecutives.** En haver d’agafar informació de Twitter, una xarxa social sense control en el seu contingut, poden haver-hi textos en els quals existeixen diverses lletres consecutives en una paraula, per facilitar l’entrenament aquestes lletres s’eliminen per tal que només en quedi una. Per exemple: “Holaaa” = “Hola”.
- **Reemplaçar emojis.** Encara que els emojis ens donen informació molt útil sobre el sentiment d’una oració, hem de convertir aquesta imatge en text que el nostre model pugui llegir. Per exemple: :) = <smile> o :(= <sad>.
- **Eliminar caràcters no alfabètics.** Caràcters com números, símbols o caràcters d’un altre alfabet no aporten cap informació útil al model i, per tant, s’han d’eliminar.

11.1.2.2. Tokenització i farciment

Tant la tokenització com el farciment, o *padding*, són tècniques NLP que ens ajuden a convertir el text llegible per un ordinador. [49]

La tokenització consisteix a separar un text en diferents parts, o tokens. Els tokens poden ser paraules, caràcters o subparaules. Un text se sol tokenitzar per espais, és a dir, cada espai separa un token.

En aquest projecte s'utilitza un algoritme anomenat Word2Vec. Aquest algoritme utilitza una xarxa neuronal per a formar associacions en un text, això ens permet donar context a una oració entre els tokens que s'han format en fer la tokenització. Aquest és un dels passos més importants en models de prediccions de textos.

En totes les xarxes neuronals és essencial que l'entrada sigui de la mateix tipus i de la mateixa mida. Malgrat això, les nostres dades varien molt en tamany, sigui de forma natural o a l'acabar el preprocessament. És per això que necessitem que tots els textos tinguin el mateix tamany i per aconseguir-ho utilitzem la tècnica de *padding* o farciment.

Aquesta tècnica consisteix en afegir *tokens* al començament i final de l'oració per incrementar el tamany fins al qual es requereix. També es pot utilitzar per decrementar el tamany.

11.1.2. Entrenament

Per tal que un model de *Machine Learning* sigui capaç de produir un comportament no arbitrari ha de ser entrenat amb unes dades concretes. Existeixen tres tipus d'entrenaments.

L'entrenament per reforç, utilitzat només en sistemes de *Deep Learning*, els quals ja no cerquen aprendre mitjançant dades externes sinó també per si mateixos. Aquest entrenament consisteix en prova i error que, finalment, arriba al comportament desitjat. Un exemple molt famós que utilitza aquest tipus d'entrenament és l'ordinador *Stockfish*, el qual no ha sigut possible de vèncer per cap Gran Mestre d'escacs. [43]

D'altra banda, l'entrenament no supervisat, no fa servir etiquetes per predir comportaments, tampoc cerca un tipus de dada específica sinó que cerca patrons o característiques que es repeteixen, però no són etiquetables. Un exemple de sistema que utilitza aquest tipus d'entrenament seria el del reconeixement facial.

Per últim, l'entrenament supervisat és el més utilitzat i el que s'utilitza en aquest projecte. Aquest tipus d'entrenament agafa dades etiquetades, en el nostre cas una frase i una puntuació del 0 al 4, i mitjançant processar aquestes dades és capaç de donar un número del 0 al 4 a qualsevol frase que hi hagi com entrada. Aquest entrenament el podem veure també en el reconeixement d'imatges en les quals per exemple hi ha l'etiqueta del que hi ha en la imatge.

Aquest projecte en fa ús d'un conjunt de dades que disposa d'1,6 milions de tweets i cadascun d'aquests tweets està etiquetat del 0 al 4, sent 0 negatiu, 2 neutra i 4 positiu. El conjunt de dades es pot accedir en: <https://www.kaggle.com/kazanova/sentiment140>. A continuació es mostra el repartiment d'entrades negatives i positives del conjunt de dades, sent mínim el nombre d'entrades neutrals.

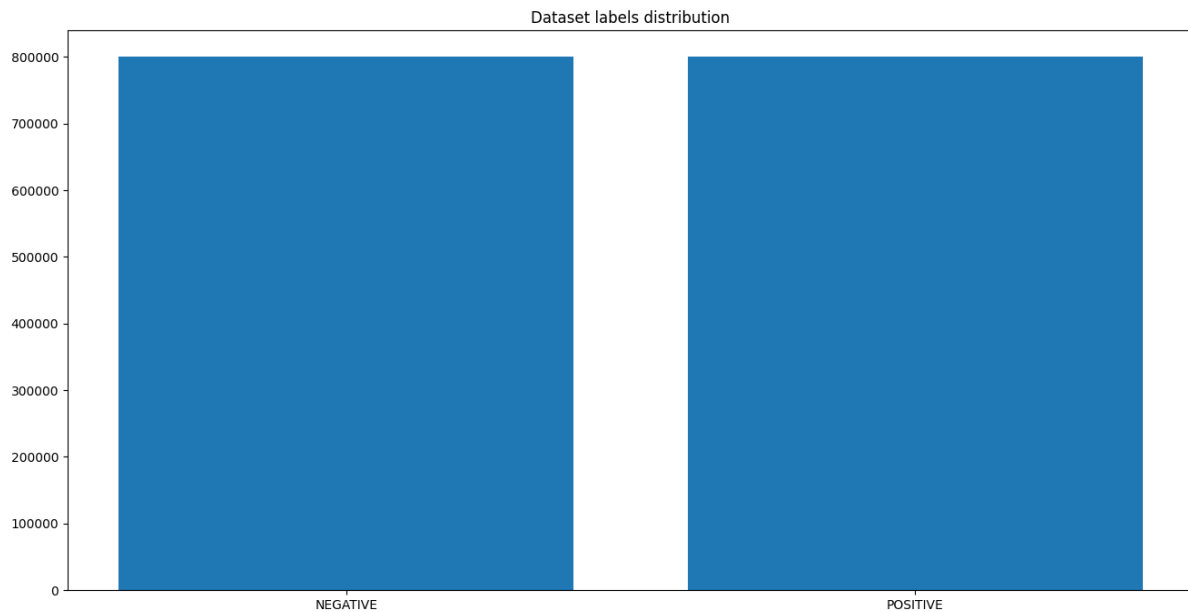


Figura 14: Distribució d'etiquetes en el dataset

11.1.3. Modelatge i resultats de l'entrenament

Una vegada sabem que model volem utilitzar per al nostre propòsit, podem començar a modelar-ho amb els coneixements adquirits en els apartats anteriors. Per començar s'utilitzarà una LSTM per tal de fer les prediccions de la forma més precisa possible. Però com acabem d'observar hem de fer uns passos abans de fer la predicció, és per això que necessitem capes addicionals a la nostra xarxa neuronal.

La primera capa és la de preprocessament de text, aquesta capa se sol anomenar *Embedding* ja que agafa totes les parts necessàries per preprocessar un text i les ajunta en una sola capa. Aquesta capa funciona tal que converteix cada paraula d'una frase en un vector d'enters únic. Després d'això, usa l'algoritme Word2Vec mencionat anteriorment que

accedeix a una taula on les claus són aquests enters i les relaciona directament amb associacions de paraules, fent ús de la tokenització [52]. En la **Figura 15** podem observar un diagrama de com funciona l'algorisme Word2Vec

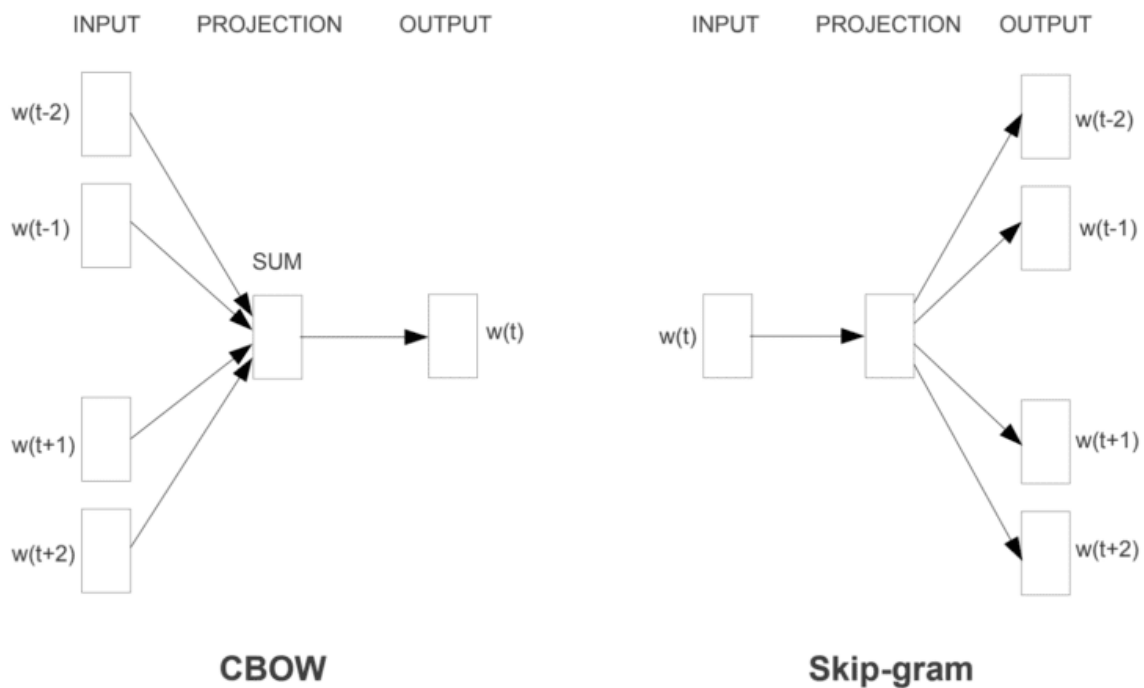


Figura 15: Representació de l'algorisme Word2Vec

La segona capa és la de *Dropout*, aquesta capa assigna 0 aleatòriament a les entrades després de cada iteració de l'entrenament, en el nostre cas fem 8 iteracions, o *epochs* [53]. Això ens serveix per evitar el sobreajustament, o *overfitting* del model, una de les bones pràctiques que veurem més endavant. *Overfitting* ocorre quan un model aprèn amb molt detall de l'entrenament, això té un impacte negatiu en l'eficiència del model, ja que el model aprèn fluctuacions o detalls que no són necessaris. [54]

La tercera capa consisteix en la LSTM, aquesta capa només introdueix el model de Long Short-Term Memory en la nostra xarxa neuronal. Per tant, ens serveix per poder introduir el funcionament d'aquest model als resultats obtinguts en les anteriors capes.

Per últim, la capa de *Dense* és una xarxa neuronal independent que s'encarrega de rebre totes les entrades de l'anterior capa i implementa la funció d'activació, en el nostre cas *sigmoid*. Aquesta capa transforma el tamany de les entrades per tal de donar un output adequat. [51]

A continuació en la **figura 16** es mostra el resultat de cridar `model.summary()` en el model creat i, per tant, podem observar amb més detall les capes de les que constitueix la nostra LSTM.

```

Model: "sequential"
-----
Layer (type)                Output Shape              Param #
-----
embedding (Embedding)       (None, 30, 300)          169601700
-----
dropout (Dropout)           (None, 30, 300)          0
-----
lstm (LSTM)                  (None, 100)              160400
-----
dense (Dense)                (None, 1)                101
-----
Total params: 169,762,201
Trainable params: 160,501
Non-trainable params: 169,601,700

```

Figura 16: Resum d'arquitectura del model

Una vegada tenim totes les capes definides, només ens queda entrenar aquest model amb el conjunt de dades que hem introduït anteriorment. Durant aquest entrenament, en cada iteració la precisió ha anat augmentant com es pot veure en la **figura 17**.

```

Epoch 1/8
1250/1250 [=====] - 518s 404ms/step - loss: 0.5228 - accuracy: 0.7354 - val_loss: 0.4611 - val_accuracy: 0.7814
Epoch 2/8
1250/1250 [=====] - 521s 417ms/step - loss: 0.4776 - accuracy: 0.7683 - val_loss: 0.4528 - val_accuracy: 0.7868
Epoch 3/8
1250/1250 [=====] - 503s 402ms/step - loss: 0.4693 - accuracy: 0.7740 - val_loss: 0.4498 - val_accuracy: 0.7890
Epoch 4/8
1250/1250 [=====] - 503s 402ms/step - loss: 0.4644 - accuracy: 0.7765 - val_loss: 0.4462 - val_accuracy: 0.7896
Epoch 5/8
1250/1250 [=====] - 504s 403ms/step - loss: 0.4631 - accuracy: 0.7779 - val_loss: 0.4456 - val_accuracy: 0.7914
Epoch 6/8
1250/1250 [=====] - 503s 403ms/step - loss: 0.4613 - accuracy: 0.7791 - val_loss: 0.4421 - val_accuracy: 0.7920
Epoch 7/8
1250/1250 [=====] - 555s 444ms/step - loss: 0.4600 - accuracy: 0.7794 - val_loss: 0.4425 - val_accuracy: 0.7917
Epoch 8/8
1250/1250 [=====] - 601s 481ms/step - loss: 0.4596 - accuracy: 0.7799 - val_loss: 0.4421 - val_accuracy: 0.7930
313/313 [=====] - 30s 95ms/step - loss: 0.4421 - accuracy: 0.7930

```

Figura 17: Precisió i perdudes del model en cada iteració

No només això sinó que la **figura 18** ens mostra d'una forma molt visual l'evolució de la precisió i les perdudes durant totes les iteracions de l'entrenament.

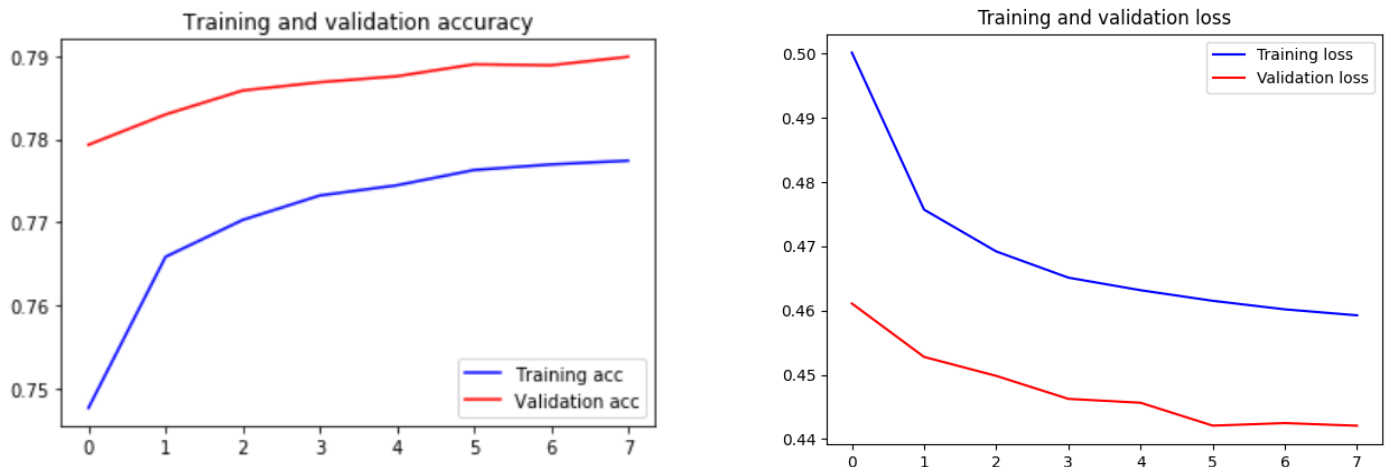


Figura 18: Evolució de precisió i perdudes durant l'entrenament

Finalment, en terminar l'entrenament, la precisió obtinguda ha sigut d'un **0,792953**, una precisió més que acceptable tenint en compte que la interpretació de frases és molt ambigua i, fins nosaltres podem interpretar una oració de forma errònia.

```

ACCURACY: 0.7929531335830688
LOSS: 0.44206830859184265

```

Figura 19: Precisió i perdudes resultants del model

11.1.4. Ús de *mlint* al model

Per tal de mantenir una bona qualitat de codi i dur a terme les millors pràctiques possibles al desenvolupar el model de *Machine Learning*, s'ha utilitzat el *linter mlint*, desenvolupat per Bart van Oort. Aquest *linter* s'utilitza per línia de comandos i s'utilitza per avaluar projectes d'intel·ligència artificial, *Machine Learning* o, fins i tot, *Deep Learning*, que han sigut duts a terme en Python. Se centra sobretot en el codi font, les dades utilitzades per l'entrenament i altres configuracions que veurem en aquest mateix apartat, fent-se eco de les bones pràctiques de *Machine Learning* definides per [SE4ML](#) i [Google's ML Rules](#).

Aquesta eina fa servir 5 *linters* diferents per tal d'assegurar la màxima qualitat de codi, aquests *linters* són *PyLint*, *MyPy*, *Black*, *isort* i *Bandit*, sent els dos primers més enfocats en

el codi i els altres 3 en l'estructura d'aquest. Gràcies a això, podem assegurar la màxima qualitat de codi. Per continuar amb la qualitat de codi, també implementa unes regles per al testatge, amb una sintaxi molt simple. Juntament amb això, una altra regla que utilitza és la d'integració continua, executant els reports dels *linters* i els pertinents tests. Per acabar, encara que hi existeixen altres regles no aplicables en aquest projecte, s'assegura de portar una bona estructura de directoris, classificant les dades per a l'entrenament, el codi del model i els tests en els seus directoris corresponents.

En la **Figura 20** es pot observar un tros d'anàlisi de *mllint* on especificar que la gestió de dependències del model és correcte i que tots els *linters* de qualitat de codi no han donat cap error, encara observant que el resum de qualitat de codi no és del 100%, ja que el *linter* de *Bandit* no ha sigut utilitzat perquè és més enfocat en la seguretat de les importacions de paquets.

Dependency Management (<code>dependency-management</code>) — 100.0%				
Passed	Score	Weight	Rule	Slug
✓	100.0%	1	Project properly keeps track of its dependencies	<code>dependency-management/use</code>
✓	100.0%	1	Project should only use one dependency manager	<code>dependency-management/single</code>
✓	100.0%	1	Project places its development dependencies in dev-dependencies	<code>dependency-management/use-dev</code>
<i>Total</i>				
✓	100.0%		Dependency Management	<code>dependency-management</code>

Code Quality (<code>code-quality</code>) — 73.9%				
Passed	Score	Weight	Rule	Slug
✓	100.0%	1	Project should use code quality linters	<code>code-quality/use-linters</code>
✓	100.0%	1	All code quality linters should be installed in the current environment	<code>code-quality/linters-installed</code>
✓	100.0%	1	Pylint reports no issues with this project	<code>code-quality/pylint/no-issues</code>
✓	100.0%	1	Pylint is configured for this project	<code>code-quality/pylint/is-configured</code>
✓	100.0%	1	Mypy reports no issues with this project	<code>code-quality/mypy/no-issues</code>
✓	100.0%	1	Black reports no issues with this project	<code>code-quality/black/no-issues</code>
✓	100.0%	1	isort reports no issues with this project	<code>code-quality/isort/no-issues</code>
✓	100.0%	0	isort is properly configured	<code>code-quality/isort/is-configured</code>
<i>Total</i>				
✓	73.9%		Code Quality	<code>code-quality</code>

Figura 20: Captura de pantalla d'un tros d'anàlisi de *mllint*

11.2. Implementació de l'aplicació web

L'aplicació web s'ha desenvolupat majoritàriament en Webstorm, un IDE molt utilitzar per al desenvolupament d'aplicacions web amb diferents frameworks. El framework utilitzat ha sigut *React*, un framework de Javascript que ens aporta dinamisme i flexibilitat a l'hora de construir una pàgina web. Aquest framework funciona amb components, cada component disposa d'una part lògica i una part gràfica, la part lògica serveix per definir el comportament d'aquest component i la part gràfica ens serveix per desplegar visualment aquest component mitjançant HTML. Aquesta arquitectura ens permet la reutilització de components per tal d'utilitzar-los en diferents pàgines, sense duplicació de codi.

Per tant de tenir una bona qualitat de codi, imprescindible en tots els aspectes de la nostra aplicació, s'hi ha fet servir el *linter*, ESLint, el qual ve integrat per defecte en Webstorm i ens ajuda a mantenir un codi sense repeticions i amb una qualitat adient. També Webstorm implementa un sistema de tests per defecte que s'executen a l'iniciar l'aplicació, aquests tests se centren només en el fet que el sistema estigui ben construït.

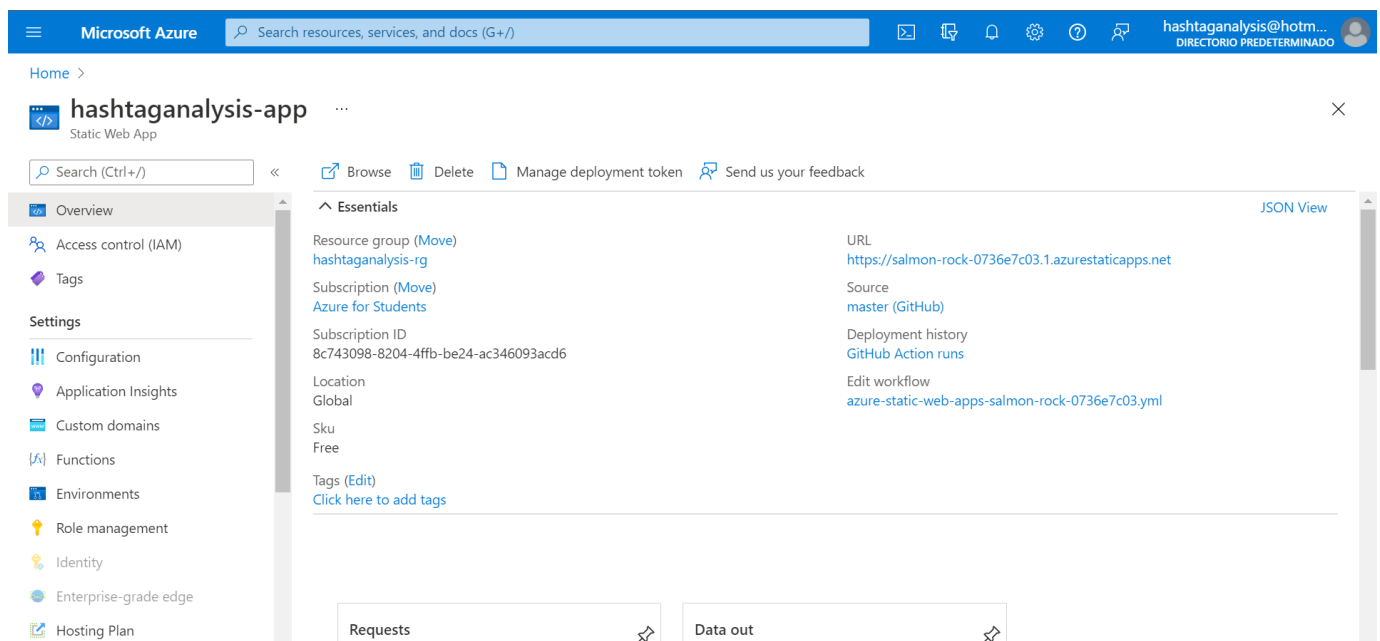
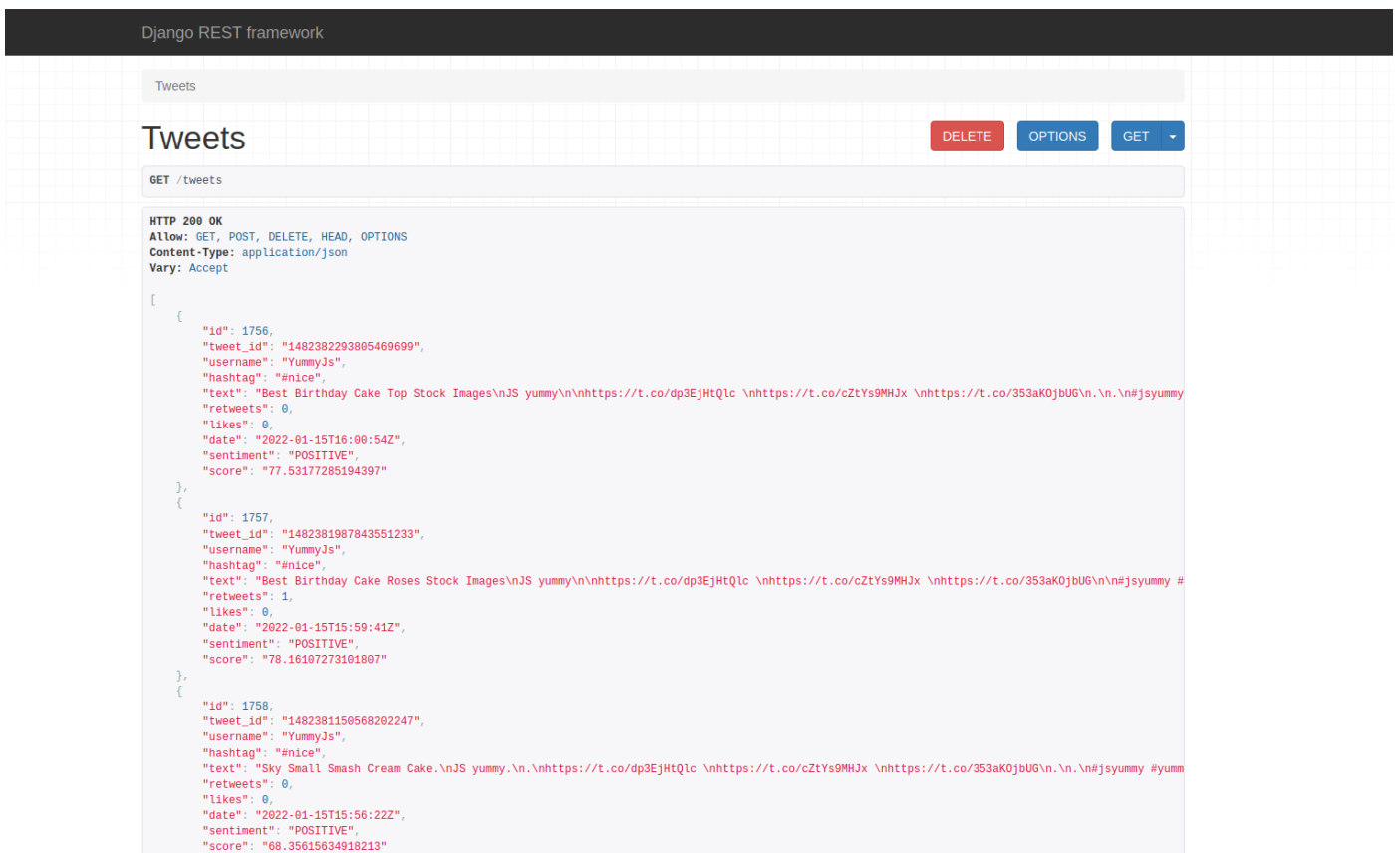


Figura 21: Entorn de desplegament Azure

11.3. Implementació del servidor

El servidor de *backend* s'ha implementat en Pycharm, un IDE utilitzat pel desenvolupament d'aplicacions en Python. El framework utilitzat és *Django*, un framework de Python que ens ofereix la possibilitat de desenvolupar una RESTful API amb una documentació molt extensa.

La REST API compta amb diversos *endpoints* que són cridats per l'aplicació web i que ofereixen la informació necessària per a emplenar la pàgina web. La informació de l'aplicació és guarda en una base de dades PostgreSQL que es gestiona de forma interna per part de *Django*, cosa que facilita el desenvolupament de migracions i models.



```
HTTP 200 OK
Allow: GET, POST, DELETE, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

[
  {
    "id": 1756,
    "tweet_id": "1482382293805469699",
    "username": "YummyJs",
    "hashtag": "#nice",
    "text": "Best Birthday Cake Top Stock Images\nJS yummy\n\nhttps://t.co/dp3EjHTQlc \nhttps://t.co/cZtYs9MHJx \nhttps://t.co/353aK0jbuG\n.\n.\n#\njsyummy",
    "retweets": 0,
    "likes": 0,
    "date": "2022-01-15T16:00:54Z",
    "sentiment": "POSITIVE",
    "score": "77.53177285194397"
  },
  {
    "id": 1757,
    "tweet_id": "1482381987843551233",
    "username": "YummyJs",
    "hashtag": "#nice",
    "text": "Best Birthday Cake Roses Stock Images\nJS yummy\n\nhttps://t.co/dp3EjHTQlc \nhttps://t.co/cZtYs9MHJx \nhttps://t.co/353aK0jbuG\n.\n.\n#\njsyummy",
    "retweets": 1,
    "likes": 0,
    "date": "2022-01-15T15:59:41Z",
    "sentiment": "POSITIVE",
    "score": "78.16107273101807"
  },
  {
    "id": 1758,
    "tweet_id": "1482381150568202247",
    "username": "YummyJs",
    "hashtag": "#nice",
    "text": "Sky Small Smash Cream Cake.\nJS yummy.\n\nhttps://t.co/dp3EjHTQlc \nhttps://t.co/cZtYs9MHJx \nhttps://t.co/353aK0jbuG\n.\n.\n#\njsyummy",
    "retweets": 0,
    "likes": 0,
    "date": "2022-01-15T15:56:22Z",
    "sentiment": "POSITIVE",
    "score": "68.35615634918213"
  }
]
```

Figura 22: Documentació d'usuaris del servidor backend

11.4. Implementació del servidor *Machine Learning*

Per tal que l'aplicació pugui fer les prediccions necessàries és vital que el nostre model estigui disponible per tal de ser utilitzat. Hi existien diferents opcions en com resoldre aquest problema. El primer enfocament va ser dirigit en guardar el model i les seves funcions per les prediccions en el mateix servidor de dades. Aquest enfocament oferia molta rapidesa a l'hora de fer les prediccions necessàries, en canvi, l'espai utilitzat per la REST API era de més de 600 MB.

Aleshores, es va decidir emmagatzemar aquest model amb una RESTful API pròpia que només contingui un *endpoint* que fes la predicció d'un text. Aquesta solució ens oferia una millora en el manteniment del sistema i separava el model de les dades. Aquesta API s'ha desenvolupat en PyCharm, com l'altra API, ja que també s'utilitza un framework de Python, més concretament utilitza *FastAPI*, una versió de *Flask*, que permet el desenvolupament d'APIs de tamany petit on la quantitat d'*endpoints* és mínima.

Aquest servidor s'ha desplegat en *DigitalOcean* utilitzant una imatge de *Docker* i el repositori de *GitHub*. Com *git* no accepta fitxers de més de 100 MB, s'ha hagut de fer servir *ifs*, una eina per guardar fitxers de més de 100 MB. *Git* substitueix aquests fitxers per punters que apunten directament al fitxer desplegat en *ifs*.

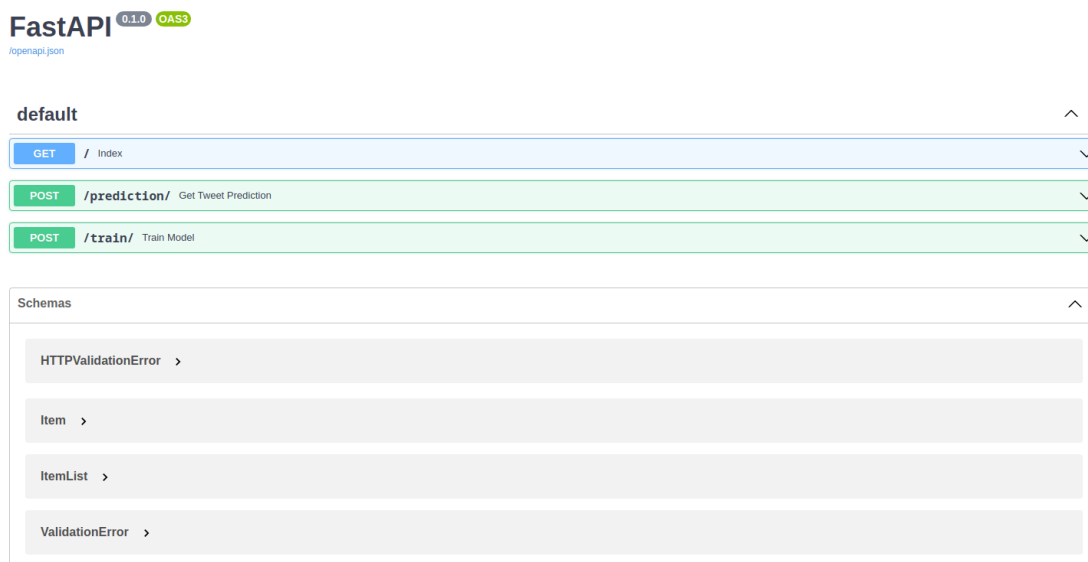


Figura 23: Documentació del servidor ML (OpenAPI)

12. Validació de requisits

Durant aquest apartat es realitzarà la validació dels requisits funcionals i no funcionals que hem descrit prèviament. La validació dels requisits funcionals es portaran a cap individualment per cada funcionalitat que s'ha especificat juntament amb una petita explicació del resultat final de la funcionalitat. Per altra banda, els requisits de qualitat es validaran una vegada terminada i testejada l'aplicació, alguns d'aquests es validaran automàticament a partir d'informació que ens ofereix el model, altres es validaran manualment testejant l'aplicació.

12.1. Validació de requisits funcionals

Per tal de validar els requisits funcionals s'han realitzat tests d'integració incrementals per a cada una de les funcionalitats, és a dir, a l'hora d'implementar una funcionalitat es realitzen les pertinents proves unitàries d'aquesta funcionalitat i, a més a més, es realitzen més proves per a funcionalitats que ja estaven testejades i implementades per tal d'assegurar el correcte funcionament d'aquestes després d'introduir la nova funcionalitat.

- **Gestió d'accés al compte.** Un usuari pot registrar-se, iniciar i terminar sessió, iniciar sessió amb Google, consultar el seu perfil, modificar-lo i eliminar-lo del sistema.
- **Gestió del compte.** Un usuari pot consultar el seu perfil, modificar el seu nom d'usuari i contrasenya, esborrar el seu compte, afegir un amic al seu compte i consultar perfils d'amics.
- **Gestió d'anàlisi de *hashtags*.** Un usuari pot generar una anàlisi buscant un *hashtag* per analitzar, interactuar, guardar aquesta anàlisi, notificar de prediccions errònies i accedir a Twitter a partir d'un tweet.
- **Visualitzar trendings.** Un usuari pot visualitzar els *trends* filtrant per país, es desplega un llistat de 10 *hashtags* o paraules claus que poden ser analitzades també.

12.2. Validació de requisits no funcionals

Per tal de validar els requisits no funcionals s'ha enviat un Google Form a 5 persones preguntant sobre l'aparença i la facilitat d'ús de l'aplicació, juntament amb un petit text on poden deixar les seves opinions personals o millores. A més a més, s'ha utilitzat l'aplicació *Postman* per mesurar el temps en realitzar l'anàlisi d'un *hashtag* o paraula clau.

Per finalitzar, per validar els requisits no funcionals de components de *Machine Learning* s'han utilitzat mètriques directament extretes del model ML, juntament amb la utilització de *mllint* per assegurar la qualitat de codi.

12.2.1. Validació de requisits no funcionals d'aplicacions web

Requisit	Resultat	Validació
Aparença	S'ha enviat un Google Form a 5 usuaris i la nota mitjana ha sigut de 3.5 sobre 5, afegint que colors menys vius donarien més contrast a l'aplicació.	Validat
Facilitat d'ús	S'ha enviat un Google Form a 5 usuaris i la nota mitjana ha sigut de 4 sobre 5, la majoria dels usuaris agraeixen la interfície senzilla que s'ha creat.	Validat
Velocitat i latència	S'ha mesurat el temps de resposta via l'aplicació <i>Postman</i> , aquestes respostes a un anàlisi d'un <i>hashtag</i> amb 100 <i>tweets</i> són, de mitjana, 7.5 segons.	Validat
Fiabilitat i disponibilitat	L'aplicació és del tot funcional qualsevol hora del dia i qualsevol dia de l'any.	Validat
Adaptabilitat	L'aplicació hi és disponible en Google Chrome, Microsoft Edge, Mozilla Firefox i Brave.	Validat

Taula 12: Taula de validacions de requisits no funcionals d'aplicacions web

12.2.2. Validació de requisits no funcionals de components de ML

Requisit	Resultat	Validació
Precisió	La precisió en la predicció del model és d'un 79.29%.	Validat
Consistència	Els resultats de precisió i perdudes donat en tots els entrenaments i els resultats de les prediccions són consistents entre ells, no baixant del mínim indicat.	Validat
Qualitat de codi	S'ha utilitzat <i>mllint</i> per al desenvolupament del model, <i>ESLint</i> per al desenvolupament de l'aplicació i <i>Pylint</i> per al desenvolupament dels servidors.	Validat

Taula 13: Taula de validacions de requisits no funcionals de components de ML

13. Bones pràctiques d'enginyeria

Hi ha projectes exitosos dels quals tots hem sentit parlar i que acaben en un interval de temps prèviament planificat. Per altra banda, hi ha altres projectes que es queden a l'ombra i que, encara tenint una idea innovadora, no arriben a bon port. Molts experts s'han preguntat que diferencia un projecte exitós d'un que no i la majoria apunten a la forma en la qual s'ha desenvolupat. Existeixen un seguit de bones pràctiques que, no assegurin sinó, ajuden al fet que un projecte sigui un èxit. Durant aquest apartat s'explicaran quines bones pràctiques s'han adoptat en aquest projecte i de quina forma s'ha fet per tal de seguir un bon desenvolupament d'aquest [33]. A continuació en la **Figura 24** es pot veure un resum categoritzat de les bones pràctiques que es detallaran durant aquest capítol.

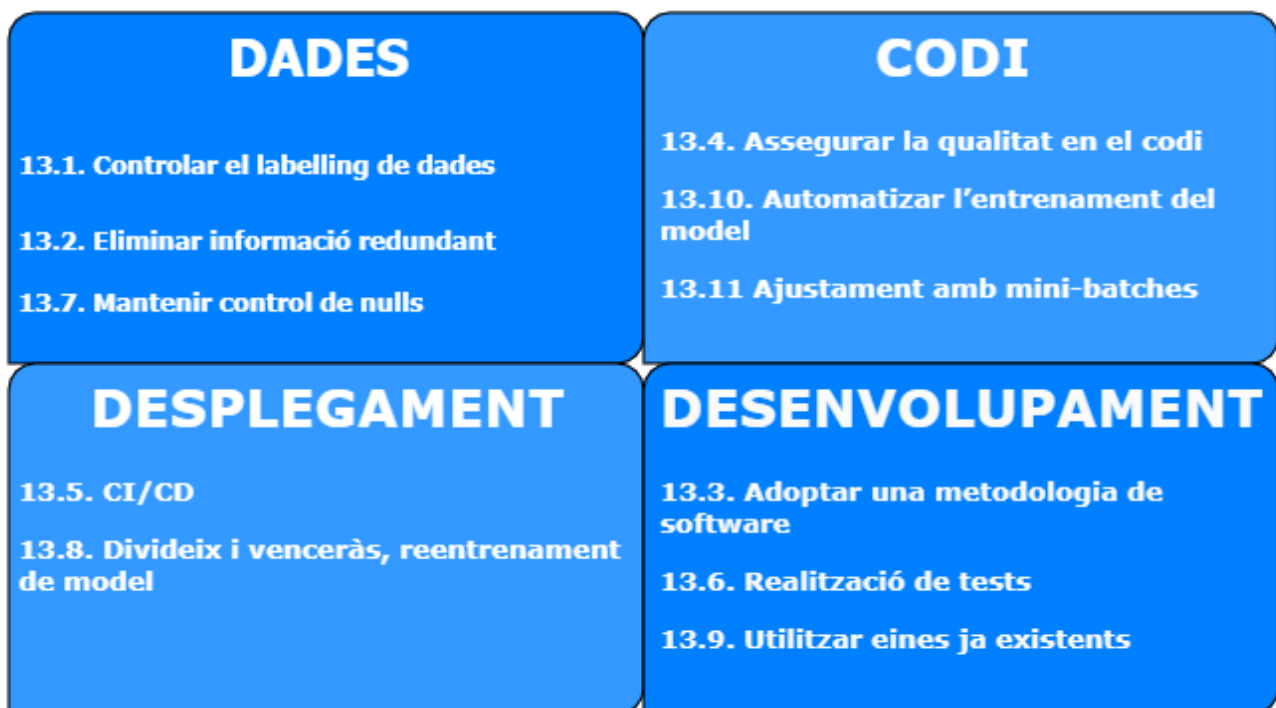


Figura 24: Bones pràctiques d'AI Engineering agrupades per categoria (Elaboració pròpia)

13.1. Controlar el *labelling* de dades

Els sistemes de *Machine Learning* es divideixen en dues grans classes, supervisat i no supervisat. La diferència entre aquests dos és que supervisat utilitza etiquetes en l'entrenament per tal que el model pugui predir el que es necessita i, per altra banda, el no supervisat no. El nostre model és supervisat, ja que necessitem etiquetes per saber si una frase és negativa, positiva o neutral i, per tant, s'ha de mantenir un control estricte d'aquestes etiquetes, pel fet que poden existir equivocacions o mal interpretacions en un model de predicció com és el nostre. Per exemple, podem tenir un conjunt de dades que identifica molt bé quan una frase és negativa a partir de tenir molts registres d'aquest cas, però que no identifica de manera igualitària quan una frase sigui neutral, ja sigui per mancances en el conjunt de dades o que els registres d'aquest cas són insuficients [32].

13.2. Eliminar informació redundant

Encara que *Twitter* és una aplicació que té com a focus expressar una opinió en el menor nombre de caràcters possible, inclòs limitant aquest nombre, en una frase pot existir informació o paraules redundants que no donen cap classe de context ni ajuden a definir la natura d'una oració. Podem posar l'exemple de la següent frase: *Ahir vaig anar a comprar i va ser una experiència dolenta*. En aquesta frase existeix informació redundant que al model no l'ajuda a predir si aquesta oració és positiva, negativa o neutral i que només elimina recursos a l'haver de computar més paraules de les degudes. Per eliminar aquest problema, hem d'introduir un diccionari de paraules redundants, és a dir, paraules que sabem no donen context a la frase, com per exemple les paraules *ahir, vaig anar, va, i, etc*. A l'eliminar aquestes paraules ens quedem amb la informació important: *comprar i ser una experiència dolenta*, que directament ens indica que l'oració té connotació negativa.

13.3. Adoptar una metodologia de software

Una metodologia de desenvolupament de *software* és un conjunt d'eines, models i normes que segueixen una filosofia de desenvolupament de programes i que s'utilitzen per a estructurar i controlar el procés de construcció d'un projecte *software*. Hi ha diferents metodologies que es poden escollir per el desenvolupament del sistema, per tant, s'ha d'escollir aquell que es coordina de millor manera, no només amb l'objectiu del projecte, sinó que també amb el personal i eines que es disposen. Per aquest projecte s'ha escollit una metodologia *Agile*, més concretament *Scrum*, una de les metodologies més emprades en tot projecte. Aquesta ens ajuda a planificar el desenvolupament del projecte i fer

entregues petites en intervals de temps per tal d'assegurar que els *stakeholders* estan d'acord amb el que s'està construint.

13.4. Assegurar la qualitat en el codi

Una vegada hem acabat la planificació i especificació del sistema i dels requeriments toca implementar el sistema. Aquesta és una de les fases on més projectes fracassen, sigui per la pobre qualitat del *software* produït o per una planificació imprecisa. Per evitar que la qualitat del nostre codi no sigui la suficient per tal que el nostre projecte sigui exitós tenim eines de control de qualitat que ens indiquen que és millorable del nostre codi. No només ens ajuda a seguir unes bones pràctiques a l'hora de programar sinó que també té múltiples beneficis posteriors, com per exemple a l'hora de debugar i encontrar errors o per reutilitzar i mantenir el sistema. En aquest projecte s'utilitzaran diferents eines per a assegurar la qualitat del codi, una d'elles és *mllint* per tal de construir un model eficient i sense errors. Aquest *linter* utilitza diferents eines per controlar la qualitat del codi, com per exemple *Pylint*, *Black*, *isort* i *Pypi* [35] [36].

13.5. CI/CD

La integració i desplegament continu és una de les millors pràctiques a l'hora de desenvolupar un projecte de *software*, ja que ens permet tenir un control major sobre el procés de desenvolupament i introduir tests que és una de les parts més importants del desenvolupament d'un projecte. En el meu cas s'utilitza GitHub Actions, un *framework* de GitHub que ens dona l'opció d'integrar directament CI/CD en el nostre repositori de codi. A més a més s'ajusta a les teves necessitats, ja que es pot personalitzar i modificar-ho al teu gust. No només ens serveix per al desplegament de la pàgina web, sinó també el del model de *Machine Learning*, es poden adoptar diferents metodologies amb la integració i el desplegament continu de models de *ML*, com per exemple desplegar el model només si s'aconsegueix una precisió mínima o si es compleixen diferents tests [30].

13.6. Realització de tests

Juntament amb la integració i el desplegament continu ens trobem amb els tests. La realització de tests és una part fonamental del desenvolupament *software*. És tan important que se sol reservar una fase únicament per realitzar tests i assegurar el correcte

funcionament del sistema. En el nostre cas utilitzarem tests per tal de mesurar que el model fa les prediccions de forma correcta i sense errors. Es realitzaran tests unitaris per ambdós comportaments, tant si una frase és positiva com negativa. Per la realització dels tests utilitzaré el linter *mllint* que disposa de regles per tal d'executar diferents tests que s'especifiquen i extreure un resultat i un *coverage*. Aquests tests s'integraran amb el CI/CD per tal de que no es faci cap desplegament que no hi hagi superat tots els tests existents [33].

13.7. Mantenir control de *nulls*

Un dels errors més freqüents a l'hora de realitzar l'entrenament d'un model d'aprenentatge automàtic és no tractar les dades que no tenen valor. Hem de tenir en compte que el nostre model ha de ser capaç de, no només predir correctament en el testatge, sinó també de predir correctament allò que l'usuari vol. Per tant, si no realitzem un correcte tractament d'aquests valors, el model no tindrà una precisió adequada. Aquest tractament es pot fer a l'hora d'extreure el conjunt de dades amb el qual entrenarà el model, com també es pot fer en el mateix codi del model. En el meu cas, les dades nul·les es tracten a l'hora d'entrenar el model amb el conjunt de dades extretes [26].

13.8. Divideix i venceràs, reentrenament de model

Sabem que per construir un model de *Machine Learning* des de zero, es necessita un conjunt de dades per tal d'entrenar aquest model i que faci el propòsit pel qual va ser desenvolupat. A més de l'enorme importància que té escollir un conjunt de dades que ens doni una precisió òptima, hem de testejar aquest model. Podem pensar a agafar un conjunt diferent de dades per tal de fer el testeig, però això ens donaria el mateix resultat sempre, ja que el conjunt de dades utilitzades pel entrenament no canvia. És per això que hem de dividir les dades inicials en dues porcions, una per entrenar i un altre per testejar. D'aquesta manera podem reentrenar el model agafant una part diferent del conjunt de dades cada vegada que sigui necessari. En aquest projecte es divideix de forma randomitzada entre un 70% i 30%, per entrenar i testejar, respectivament [26].

13.9. Utilitzar eines ja existents

Per aquesta bona pràctica hi existeix un refrany molt famós que explica exactament com hem d'afrontar un repte complex: "No reinventar la roda". Encara que l'aprenentatge

automàtic conviu amb nosaltres des de fa un temps, el que passa dins d'un model com el que estem utilitzant es tracta com una *black box*, ja sigui per la complexitat dels conceptes que s'utilitzen perquè funcioni o simplement perquè no els necessitem conèixer. No fa falta que construïm cada part d'una xarxa neuronal perquè ja existeixen eines que fan aquest treball per nosaltres. Això ens ajuda a estalviar temps que podem aprofitar per millorar el projecte i per confiar en el fet que el model generat és correcte. En aquest projecte he fet ús de *TensorFlow*, una plataforma *open source*, que ens ajuda i guia en la construcció d'un model de *Machine Learning* [25].

13.10. Automatitzar l'entrenament del model

Un model de *Machine Learning* que tingui una precisió adequada per tal d'aconseguir el comès per al qual està dissenyat necessita una quantitat aclaparadora de dades per tal de realitzar un entrenament que ens doni aquesta precisió esperada. Molt sovint, aquestes dades superen els milions de files i l'entrenament pot arribar a tardar més del que ens agradaria. Encara que no podem fer res per tal que el temps de l'entrenament sigui inferior si podem automatitzar l'entrenament per tal que no hàgim de repetir manualment el procés en cas que la precisió no sigui l'esperada. Per tal d'obtenir això podem entrenar el model amb uns paràmetres que creiem adients i en cas de no superar la precisió esperada, repetir el procés automàticament canviant els paràmetres de la xarxa neuronal sense necessitat que nosaltres la configurem manualment.

13.11. Ajustament amb mini-batches

Quan es depura un model, sovint és útil fer proves ràpides per veure si hi ha algun problema important amb l'arquitectura del mateix model. En particular, per tal d'assegurar-se que el model es pot entrenar correctament, es passa un mini-batch dins de la xarxa neuronal per veure si hi pot sobreposar-se. Si no és possible, vol dir que el model és massa complex o no és prou complex com per a adaptar-se fins i tot a un mini-batch, i molt menys un conjunt d'entrenament de mida normal. Això ens permet visualitzar si l'arquitectura del model és adient sense necessitat d'haver d'entrenar el model amb tot el conjunt de dades, cosa que pot arribar a trigar una quantitat de temps major a la ideal.

14. Discussions

Durant aquest apartat es realitzaran les pertinents discussions sobre l'informe de sostenibilitat realitzat, els resultats de la planificació prèviament definida en l'**apartat 4** i els costos finals resultants.

14.1. Informe de sostenibilitat

A continuació es realitzarà una reflexió sobre el grau de sostenibilitat en projectes de *software*, una avaluació sobre l'aprenentatge de la sostenibilitat durant la carrera i un informe en l'impacte de la mateixa en aquest projecte.

14.1.1. Autoavaluació

Durant el transcurs de la carrera se'ns ha posat a disposició moltes conferències, entrevistes, documentals i vídeos sobre la sostenibilitat en el nostre àmbit laboral. En cada assignatura ha sigut molt important tenir en compte en què pot afectar en els àmbits social, econòmic i ambiental als diferents projectes i treballs que hem de dur a terme al llarg de la nostra vida laboral.

En completar l'enquesta proposada pels responsables de GEP he aconseguit autoavaluar-me sobre els coneixements que he anat adquirint durant aquests anys, ja sigui fets que ja estan molt interioritzats en el meu dia a dia, com altres fets que no he treballat en profunditat i que és recomanable millorar.

Entre aquests coneixements crec que soc capaç d'identificar certs riscos sobre la sostenibilitat en un projecte, com també soc capaç d'adoptar tècniques per tal d'evitar aquests riscos i millorar la sostenibilitat d'un projecte informàtic.

Per altra banda, m'he adonat compte que a l'hora de respondre les preguntes de l'enquesta sobre mesures de sostenibilitat, tenia pocs coneixements. Crec que introduir mesures de sostenibilitat en un projecte és un dels factors més importants per obtenir una sostenibilitat adequada.

És per això que, encara tenint els coneixements necessaris per desenvolupar un projecte informàtic de forma sostenible, és crucial millorar en la introducció d'indicadors i mesures per tal de tenir una visió més precisa de la sostenibilitat d'un projecte.

A continuació es respondran un seguit de preguntes sobre la sostenibilitat en les dimensions ambiental, social i econòmica que ens ajudaran a aprofundir aquests conceptes en relació amb el nostre projecte:

14.1.2. Fita inicial

14.1.2.1. Dimensió ambiental

En relació amb PPP, s'ha estimat el impacte ambiental que tindrà la realització del projecte? T'has plantejat minimitzar l'impacte, per exemple, reutilitzant recursos?

El nostre projecte és una solució *software*, és a dir, no és un producte físic, per tant, no tindrà un impacte important en la dimensió ambiental. No obstant això, és necessari tenir present els costos d'electricitat i internet, que encara sent petits, afecten directament a l'ambient i que, en conseqüència, han sigut estimats.

Com he mencionat anteriorment en l'apartat d'identificació de costos *hardware*, utilitzo dos ordinadors diferents per dur a terme tasques independents. Es podria minimitzar l'impacte ambiental utilitzant només un ordinador afectant, però, a la productivitat.

En que millorarà ambientalment la teva solució a les existents?

A l'utilitzar bones pràctiques d'enginyeria hi haurà un mínim de consum de recursos, com electricitat o potència computacional que altres solucions no tenen en compte.

14.1.2.2. Dimensió econòmica

En relació amb PPP, has estimat el cost de la realització del projecte (recursos humans i materials)?

Si s'ha estimat el cost de la realització del projecte. S'ha tingut en compte el personal necessari pel desenvolupament del projecte de forma exitosa, com també s'ha identificat el diferent equip, tant *software* com *hardware*, que s'utilitzarà per treballar. A partir d'aquests costos s'ha generat un pressupost total del projecte tenint en compte possibles imprevistos i contingències.

En relació amb vida útil, com es resol actualment el problema que vols abordar (estat de l'art)? En que millorarà economicament la teva solució a les existents?

El problema que es vol abordar és la no existència del servei que volem posar a disposició de forma gratuïta, encara sent obligatori l'arribada de beneficis per finançar el desenvolupament d'aquest projecte, s'utilitzaran anuncis que ens donaran aquest benefici i ens permet continuar donant aquest servei de forma gratuïta.

La meva solució millorarà, com he dit anteriorment, en oferir un servei gratuït que les altres solucions no ofereixen o ofereixen a partir d'un pagament.

14.1.2.3. Dimensió social

En relació al PPP, que creus que t'aportarà a nivell personal la realització del projecte?

Crec que la realització d'aquest projecte m'aportarà de forma molt significativa a escala personal, ja sigui per haver d'aprendre noves tecnologies i haver de reinventar-se, com per organitzar i seguir una planificació prèvia, a més d'un informe de pressupost i sostenibilitat com el que s'està fent.

En relació a la vida útil, existeix una necessitat real del projecte?

No existeix cap empresa ni pàgina web que ofereix el servei que estem disposats a oferir de forma gratuïta per a qualsevol usuari. És per això que és totalment necessari el desenvolupament d'aquest projecte i obrirà noves portes a aquells usuaris que estiguin interessats en el nostre servei i que per raons diferents mai han utilitzat.

14.1.3. Fita final

14.1.3.1. Dimensió ambiental

En relació amb el PPP, has quantificat l'impacte ambiental de la realització del projecte? Quines mesures has pres per reduir l'impacte? Has quantificat aquesta reducció?

Sí, el cost s'ha estimat en kWh. S'ha extret directament el consum d'una factura d'electricitat de casa i eliminant els promitgos de consum de televisió, aire condicionat, etc, s'ha estimat

que el cost en kWh de dos portàtils, els quals no es treballen paral·lelament, és de 0,72 kWh, tenint en compte que li hem dedicat 540 hores al projecte, el total seria de 388,8 kWh.

A més a més hem de tenir en compte el consum dels dos servidors de Heroku i Azure que només s'activaran en el procés final del projecte i quan se li donin ús per tal de reduir l'impacte total, dedicant-li unes 50 hores a cadascun, suposant que un servidor promig consumeix 2 kWh. En total, el projecte ha tingut un cost de 638,8 kWh.

En relació amb el PPP, si fessis de nou el projecte, podries fer-ho amb menys recursos?

Existeix la possibilitat d'eliminar el servidor de ML, integrant el model dins del *backend*, això encara reduint el consum energètic i augmentant l'eficiència, tindria un cost econòmic addicional a l'utilitzar un servidor que requereix un espai molt major al que es disposa en la prova gratuïta. En cas de tenir un pressupost, el qual es pot utilitzar per despeses d'arquitectura, si es tornés a fer el projecte es podria eliminar un servidor de l'arquitectura i utilitzar menys recursos.

En relació amb la vida útil, quins recursos estimes que s'usaran durant la vida útil del projecte? Quin serà l'impacte ambiental d'aquests recursos?

Al ser un projecte de *software* els recursos que s'utilitzaran seran energètics, és a dir, majoritàriament de servidors. Encara que l'impacte serà mínim, ja que l'aplicació no és molt gran, es requerirà energia elèctrica durant la vida útil del projecte.

En relació amb la vida útil, el projecte permetrà reduir l'ús d'altres recursos? ¿Globalment, l'ús del projecte millorarà o empitjorarà la petjada ecològica?

Aquest projecte permetrà reduir el consum d'energia que s'utilitzen en altres alternatives que s'han pogut observar al començament d'aquest document. Aquestes aplicacions alternatives són més grans i, per tant, utilitzen més recursos energètics. Respecte a la petjada ecològica, el consum energètic d'aquest projecte és mínim i, en conseqüència, no tindrà un impacte significatiu.

En relació amb els riscos, podrien produir-se escenaris que fessin augmentar la petjada ecològica del projecte?

L'únic escenari en el qual es pugui augmentar la demanda d'energia en els servidors és si l'aplicació fos utilitzada per molts usuaris, amb tot i això, l'impacte no seria significatiu.

14.1.3.2. Dimensió econòmica

En relació amb el PPP, has quantificat el cost (recursos humans i materials) de la realització del projecte? Quines decisions has pres per reduir el cost? Has quantificat aquest estalvi?

S'ha quantificat el cost de recursos humans i materials d'aquest projecte. Els recursos humans són inexistents, ja que només hi ha hagut un desenvolupador en la realització del projecte. En quant als recursos materials, s'ha obtingut una amortització total de 232,5 € entre els dos portàtils que s'han utilitzat.

Per tal de reduir costos, s'ha assegurat la utilització de servidors gratuïts, tenint en compte els requeriments d'ús. Per tant, el cost total de recursos humans i materials és de 232,5 €.

En relació amb el PPP, s'ha ajustat el cost previst al cost final? Has justificat les diferències (llicències apreses)?

Encara que el cost previst en el començament del treball tenia en compte la contractació de lloguer d'oficina, desenvolupadors, enginyers i servidors de pagament. Al cap i a la fi, aquests costos no s'han realitzat i els costos de portàtils s'ha ajustat el que s'havia previst.

En relació amb la vida útil, quin cost estimes que tindrà el projecte durant la seva vida útil? Es podria reduir aquest cost per fer-lo més viable?

En cas que el projecte es popularitzi, seria necessari la contractació de personal per a fer manteniments de l'aplicació i desenvolupar millores. En aquest cas, seria necessari equip informàtic per als desenvolupadors. En resum:

Equip informàtic	Cost	Amortització
MSI Modern 14 B10RBSW (Portàtil)	869	142,2

Rol	Salari brut (€/h)	Cost brut (Salari brut * 1.3) (€/h)
Programador <i>frontend</i>	20,5	26,65
Programador <i>backend</i>	22	28,6
Dissenyador UI/UX	14,3	18,6

Programador becari	9	11,7
--------------------	---	------

Tenint en compte el salari de cada persona de l'equip i la necessitat de tenir 4 equips informàtics:

Recurs	Cost total(€)
Sou de desenvolupadors	7956,15 *
Equip informàtic	568,8

* Cost mensual

En relació amb la vida útil, s'ha tingut en compte el cost dels ajustos / actualitzacions / reparacions durant la vida útil del projecte?

Sí, s'han tingut en compte possibles ajustos, actualitzacions o manteniment del sistema. En l'equip de desenvolupadors introduït anteriorment, es disposa de totes les parts per part d'implementar qualsevol millora o reparació, sigui de *backend*, *frontend* o UI/UX.

En relació amb els riscos, podrien produir-se escenaris que perjudiquen la viabilitat del projecte?

El projecte podria perdre viabilitat en cas que no fos el prou popular o els usuaris no utilitzessin l'aplicació.

14.1.3.3. Dimensió social

En relació amb el PPP, la realització d'aquest projecte ha implicat reflexions significatives a nivell personal, professional o ètic de les persones que han intervingut?

Sí, durant el desenvolupament del projecte i mentre anava avançant en l'aplicació he pogut observar quant de difícil és interpretar el sentiment d'una frase qualsevol sense tenir cap mena de context, inclòs per a una persona humana. És per això que m'he esforçat especialment en la precisió del model perquè doni les prediccions correctes la majoria de les vegades. És molt freqüent veure en xarxes socials persones discutint per poc entesa entre elles o per interpretacions dolentes, per tant, les prediccions d'aquesta aplicació havien de ser correctes per tal d'evitar aquest tipus de malentesos.

En relació amb la vida útil, qui es beneficiarà de l'ús del projecte? Hi ha algun col·lectiu que es pot veure perjudicat pel projecte? En quina mesura?

Aquelles persones usuàries de Twitter amb curiositat sobre el sentiment d'un trend o d'un *hashtag* sortiran molt beneficiades, ja que es donarà una resposta a les seves necessitats. També aquelles empreses que volen analitzar els seus propis *hashtags* o paraules claus per veure la recepció en Twitter.

No hi hauria cap col·lectiu perjudicat per aquest projecte, ja que és merament informatiu.

En relació amb la vida útil, en quina mesura soluciona el projecte el problema plantejat inicialment?

Aquest projecte soluciona el problema plantejat inicialment oferint una aplicació a qualsevol usuari per tal de poder analitzar qualsevol *hashtag* o paraula clau de Twitter sense cap mena de pagament.

En relació amb els riscos, podrien produir-se escenaris que fesin que el projecte fos perjudicial per a algun segment particular de la població?

No hi ha cap escenari en el qual aquest projecte podria ser perjudicial per a cap segment particular de la població.

En relació amb els riscos, podria crear el projecte algun tipus de dependència que deixés als usuaris en posició de debilitat?

Aquest projecte no ofereix cap classe de servei que pugui donar a lloc a alguna mena de dependència als usuaris que la utilitzin.

14.2. Resultats de la planificació

Una vegada hem acabat el desenvolupament del projecte, hem de reflectir en els canvis entre la planificació inicial que es va realitzar en GEP i la planificació resultant. Durant el transcurs del desenvolupament del projecte han aparegut diversos obstacles que no s'havien tingut en compte, com també s'han fet avenços que s'havien estimat amb un pes superior al final.

En aquest apartat s'especificaran els objectius que han sigut complerts de la planificació inicial, juntament amb una explicació del resultat obtingut, i una anàlisi dels costos finals del desenvolupament de l'aplicació.

14.2.1. Desenvolupament de les iteracions

En aquest apartat s'explicarà com han evolucionat les tasques en cada *sprint* realitzat al llarg del desenvolupament del projecte.

14.2.1.1. Planificació

Objectius proposats:

Especificar l'abast i la contextualització del projecte.

Realitzar una planificació inicial del desenvolupament del projecte.

Definir el pressupost inicial del projecte

Entrega de document final de Gestió de Projectes (GEP)

Resultats:

Especificar l'abast i la contextualització del projecte. L'objectiu va ser assolit satisfactòriament. Es va realitzar un abast del projecte, és a dir, definir els objectius del projecte, els *stakeholders* i els possibles obstacles. També es va definir la motivació, alternatives existents i la solució escollida a partir de l'anàlisi de mercat.

Realitzar una planificació inicial del desenvolupament del projecte. L'objectiu va ser assolit satisfactòriament. Es va realitzar una planificació inicial i temporal de les tasques per aconseguir els objectius del projecte.

Definir el pressupost inicial del projecte. L'objectiu va ser assolit satisfactòriament. Es va definir un pressupost inicial del projecte tenint en compte les tasques definides, el personal necessari per desenvolupar el projecte i els materials que s'utilitzaran.

Entrega de document final de Gestió de Projectes (GEP). L'objectiu va ser assolit satisfactòriament. Es va realitzar una entrega i integració del document final a partir dels resultats produïts als anteriors objectius.

14.2.1.2. Incepció

Objectius proposats:

Definir funcionalitats del projecte.

Definir requisits de qualitat.

Fer estudi de bones pràctiques.

Resultats:

Definir funcionalitats del projecte. L'objectiu va ser assolit satisfactòriament, encara que es van anar afegint funcionalitats interessants a mesura avançava el desenvolupament de les definides.

Definir requisits de qualitat. L'objectiu va ser assolit satisfactòriament, es van anar especificant requisits de qualitat tant de sistemes *software*, tant de sistemes basats en intel·ligència artificial.

Fer estudi de bones pràctiques. L'objectiu no va ser assolit en la data límit definida, aquest estudi es va anar desenvolupant a mesura que avançava l'aplicació i es descobriren noves bones pràctiques que podien ser aplicades en el nostre sistema.

14.2.1.3. Primera iteració

Objectius proposats:

Investigació sobre models de *Machine Learning*.

Construcció del model de *Machine Learning*.

Integració de *mllint* al model.

Integració del model amb l'API de Twitter.

Resultats:

Investigació sobre models de *Machine Learning*. L'objectiu va ser assolit satisfactòriament, es va investigar sobre tots els tipus de models de *Machine Learning*, adients per al nostre sistema i es va escollir LSTM a partir d'aquesta investigació.

Construcció del model de *Machine Learning*. L'objectiu va ser assolit satisfactòriament en el temps estipulat, es va construir i entrenar el model de *Machine Learning* de forma correcta, donant les prediccions adequades i esperades sense cap tipus d'obstacle o problema.

Integració de *mllint* al model. L'objectiu no va ser acabat per al final de la primera iteració i es va decidir continuar a la segona iteració, *mllint* va resultar ser una eina clau en l'aplicació de bones pràctiques del model i va portar més temps de l'esperat per totes les funcionalitats de les quals disposava.

Integració del model amb l'API de Twitter. L'objectiu va ser assolit de forma satisfactòria i sense cap dificultat, es va utilitzar l'API de Twitter que ja havia utilitzat en assignatures passades, per tant, en la integració no va sorgir cap obstacle. Tot i això, es van realitzar alguns canvis en la tercera iteració.

14.2.1.4. Segona iteració

Objectius proposats:

Realitzar l'especificació del sistema.

Construcció de la pàgina web.

Log in dels usuaris.

Registre dels usuaris.

Esborrar un usuari.

Consultar el perfil d'un usuari.

Resultats:

Realitzar l'especificació del sistema. L'objectiu va ser assolit de forma parcial, es va iniciar l'especificació del sistema encara que no es va acabar i es va decidir completar-la en la tercera iteració.

Construcció de la pàgina web. L'objectiu va ser assolit satisfactòriament sense cap dificultat, es va decidir utilitzar tecnologies familiars i que complementaven de forma correcta al que es vol construir.

Gestió d'usuaris. L'objectiu no va ser assolit, es va decidir donar més pes a la generació de l'anàlisi i la funcionalitat principal de l'aplicació abans de desenvolupar la gestió

d'usuaris. Per tant, la gestió d'usuaris es va decidir implementar en la quarta iteració, una vegada la generació d'anàlisi i gestió de *hashtags* estigués completa.

Integració de *mlint* al model. L'objectiu va ser assolit de forma satisfactòria, es va poder integrar totes les funcionalitats de *mlint* al model sense cap dificultat.

Generació de l'anàlisi. Es va decidir començar amb la generació de l'anàlisi, també es va decidir eliminar la integració del model amb l'API de Twitter i integrar-la amb el servidor de l'aplicació per tal alliberar de feina al model.

14.2.1.5. Tercera iteració

Objectius proposats:

Integració del model a la pàgina web.

Generació de l'anàlisi

Guardar i compartir una anàlisi

Resultats:

Integració del model a la pàgina web. L'objectiu va ser assolit de forma satisfactòria, es va integrar el model a la pàgina web a partir d'una petita API. L'aplicació fa trucades a l'API per fer prediccions de forma correcta.

Generació de l'anàlisi. L'objectiu va ser assolit satisfactòriament. El sistema genera una anàlisi a partir de la cerca d'un *hashtag*. Aquest objectiu va ser assolit sense cap dificultat addicional.

Guardar i compartir una anàlisi. L'objectiu no va ser assolit. La càrrega de treball de la generació de l'anàlisi i la pàgina on desplegar aquesta anàlisi va ser més gran a l'esperada i es va decidir moure aquest objectiu cap a la quarta iteració.

Accedir al tweet original. Es va decidir implementar aquesta tasca de la quarta iteració, ja que s'implementarà amb la pàgina de generació d'anàlisi i la càrrega de treball és molt petita.

14.2.1.6. Quarta iteració

Objectius proposats:

Notificar d'error en una predicció
Interacció amb els gràfics produïts
Accedir al Tweet original.

Resultats:

Notificar d'error en una predicció. L'objectiu va ser assolit satisfactòriament sense cap problema. Podem notificar d'un error en la predicció del tweet clicant un botó al costat del tweet.

Interacció amb els gràfics produïts. L'objectiu va ser assolit satisfactòriament sense cap dificultat. El gràfic produït a l'anàlisi és interactiu.

Accedir al Tweet original. Aquest objectiu es va desenvolupar en la tercera iteració com s'ha mencionat anteriorment.

Gestió d'usuaris. Aquest objectiu estava planificat inicialment de ser desenvolupat en la segona iteració, com hem mencionat anteriorment. Durant aquesta iteració s'ha desenvolupat aquest objectiu satisfactòriament, un usuari pot iniciar sessió, registrar-se modificar la seva contrasenya i el seu *about* i visualitzar el seu perfil.

Guardar i compartir anàlisi. Aquest objectiu, originalment de la segona iteració, no es va assolir per la càrrega de treball que va portar la gestió d'usuaris. Al ser dues funcionalitats senzilles d'implementar es va decidir moure, una altra vegada, aquest objectiu cap a la quinta iteració.

14.2.1.7. Quinta iteració

Objectius proposats:

Mostrar els *hashtags* més populars.
Canviar país.
Milliores d'interfície i rendiment.

Resultats:

Mostrar els *hashtags* més populars. Aquest objectiu es va assolir de forma satisfactòria. Es va crear una nova pàgina dins de l'aplicació on un usuari pot visualitzar els *hashtags* més populars amb uns filtres opcionals.

Canviar país. Aquest objectiu es va assolir de forma satisfactòria de forma paral·lela amb l'anterior objectiu, ja que entra dins de la implementació pensada. Es van desenvolupar el filtre de països per als *trendings*.

Millores d'interfície i rendiment. Aquest objectiu va ser assolit de forma satisfactòria. Es va millorar la interfície fent-la més *user-friendly*.

Guardar i compartir anàlisi. Aquest objectiu va ser assolit parcialment, es va poder implementar la funcionalitat de guardar un anàlisi en el teu perfil de forma satisfactòria. Per altra banda, l'objectiu de compartir un anàlisi no va ser assolit per falta de temps.

16.2.1.8. Sexta iteració

Objectius proposats:

Preparació de la defensa.

Resultats:

Preparació de la defensa. Aquest objectiu es va assolir de forma satisfactòria. Es va decidir acurtar aquesta iteració una setmana, per tal de finalitzar el desenvolupament de la pàgina web, tan funcionalitats que estaven inacabades com testeigos. Es va definir una guia a seguir durant la defensa, un powerpoint per acompanyar l'explicació i una demo per tal de demostrar el funcionament de la pàgina web i les seves funcionalitats.

14.2.2. Costos finals

Al principi d'aquest document es van mostrar una estimació de costos, és necessari matisar que aquesta estimació va ser pensada assumint que el desenvolupament del sistema fos realitzat per professional. Dit això, no és una estimació real, ja que l'únic desenvolupador d'aquest projecte he sigut jo mateix, és per això que en aquest apartat es realitzarà una estimació dels costos finals d'aquest projecte.

Com podem observar en la **Taula 14**, s'ha introduït només els costos de recursos que s'han utilitzat per a desenvolupar aquest projecte, és a dir els costos reals. Ambdós portàtils, un per a la realització del document i l'altre, amb més potència computacional, per al desenvolupament i pertinent entrenament del model de *Machine Learning*, no només això sinó també s'ha tingut en compte la utilització d'electricitat, extreta directament d'una factura i eliminant altres recursos no utilitzats, tenint en compte que no s'utilitzen ni es carreguen ambdós portàtils a la mateixa vegada.

Tipus de cost	Cost	Amortització
HP Laptop 14s-dq2000ns (Portàtil)	549	89,83
MSI Modern 14 B10RBSW (Portàtil)	869	142,2
Electricitat	0,1142 €/kWh	3,81
TOTAL		235,94

Taula 14: Taula de costos finals i amortització

15. Conclusions i treball futur

Ara que estem arribant a la finalització del projecte és hora de fer retrospectiva d'aquests darrers mesos i treure algunes conclusions, ja sigui de com ha sigut l'experiència, que hem après, que dificultats han aparegut durant el transcurs del projecte, possibles millores o funcionalitats per implementar en un futur i una valoració personal del treball fet.

15.1. Conclusions del projecte

Una vegada finalitzat el projecte, podem dir que s'han pogut implementar totes les funcionalitats inicialment definides. S'ha seguit una metodologia *Agile* durant el transcurs de tot el projecte que hem dividit per *sprints*. A continuació es pot veure un resum dels objectius aconseguits, fent referència directament als objectius definits prèviament, més detalladament en el **capítol 1.5.1**.

- **Objectius O1 i O2.** S'ha desenvolupat i desplegat una aplicació web amb el seu corresponent servidor per tal d'oferir les funcionalitats descrites a un usuari. Aquestes funcionalitats funcionen tal com es van definir.
- **Objectiu O3.** S'ha desenvolupat i desplegat un component *Machine Learning* per tal d'implementar *sentiment analysis* en la nostra aplicació. També s'ha desenvolupat un servidor independent de l'aplicació per tal d'utilitzar aquest model de forma eficient.
- **Objectiu O4.** S'han utilitzat bones pràctiques d'enginyeria a l'hora de desenvolupar tots els components de l'aplicació web, ja sigui la part de *frontend*, com el *backend*, com el component de *Machine Learning*.

A més, s'han implementat totes les funcionalitats que s'havien descrit prèviament tal com s'esperava i s'han complert tots els requisits no funcionals que havien sigut proposats. S'ha realitzat una planificació de les tasques del projecte per tal d'implementar les funcionalitats pertinents i s'han fet els canvis oportuns per tal de desenvolupar-les sense cap tipus de problema. També s'ha fet un estudi de mercat i una anàlisi de costos del desenvolupament d'aquesta aplicació.

15.2. Justificació de competències

En aquest apartat s'explicaran les competències escollides al principi del projecte per desenvolupar en el transcurs d'aquest, així mateix es justificaran aquestes competències descrivint com s'han desenvolupat cadascuna d'elles.

CES1.1: Desenvolupar, mantenir i avaluar sistemes i serveis software complexos i/o crítics. [En profunditat]

Aquesta competència ha sigut desenvolupada a l'hora de construir tot el sistema per a l'aplicació, ja sigui la part visual de l'aplicació, com el servidor backend i el model de *Machine Learning*. No només això, sinó que durant el desenvolupament d'aquests ha sigut necessari mantenir-los i avaluar-los per al seu correcte funcionament.

CES1.2: Donar solució a problemes d'integració en funció de les estratègies, estàndards i tecnologies disponibles. [En profunditat]

Per tal d'integrar un sistema que consta amb 3 parts diferents, aplicació web, servidor i model ML, ha sigut necessari fer una investigació sobre les tecnologies disponibles i les estratègies per tal d'integrar aquests 3 components. A més a més, s'han seguit un conjunt de bones pràctiques per tal de complir amb els estàndards en el desenvolupament software.

CES1.7: Controlar la qualitat i dissenyar proves en la producció del software. [Bastant]

Tots els components del sistema, ja sigui el model de *Machine Learning*, frontend o backend disposen de sistemes de control de qualitat anomenats *linters*, a més a més de proves en les prediccions per tal d'aconseguir la màxima precisió possible. Encara sent "bastant" la profunditat establerta prèviament, aquesta competència s'ha desenvolupat "en profunditat".

CES2.1: Definir i gestionar els requisits d'un sistema software. [En profunditat]

Aquesta competència ha sigut desenvolupada al començament del projecte on s'havien d'especificar els requisits funcionals i no funcionals del nostre sistema.

15.3. Integració de coneixements

Durant el transcurs de la carrera he anat aprenent i adquirint habilitats en diferents tècniques i tecnologies essencials per al desenvolupament d'aquest projecte.

Per començar, la gestió del projecte ha sigut una de les parts més importants, ja que una bona planificació i gestió és crítica en el desenvolupament de qualsevol projecte, sigui de *software* o no. Per aquesta part, els coneixements adquirits en **Gestió de Projectes de Software (GPS)**, **Projecte d'Enginyeria de Software (PES)**, **Conceptes de Sistemes d'Informació (CSI)** i **Empresa i Entorn Econòmic**, han sigut principals en el començament del projecte i la part corresponent a **Gestió de Projectes (GEP)**. Més detalladament, fer un estudi de mercat, observar les alternatives existents i preparar una solució d'acord amb els resultats extrets. També, adoptar una metodologia adequada durant el desenvolupament del projecte, tenint en compte els riscos i obstacles que poden sorgir. Per últim, fer estimacions sobre els costos d'un projecte, el personal necessari per dur-ho a terme en funció de temps i diners.

Per altra banda, en la part d'especificació i disseny del sistema han sigut essencials els coneixements adquirits en **Enginyeria de Requisits (ER)**, **Arquitectura del Software (AS)**, **Introducció a l'Enginyeria del Software (IES)**, **Projecte d'Enginyeria de Software (PES)**, **Gestió de Projectes de Software (GPS)**. En especial a l'hora de definir els requisits funcionals i no funcionals d'un sistema, dissenyar un diagrama UML representant del sistema, identificar casos d'ús, especificar una arquitectura adequada per al sistema i fer ús de patrons en la implementació del sistema.

Per últim, en la part d'implementació del sistema moltes assignatures han sigut importants en termes de conceptes i coneixements adquirits, **Aplicacions i Serveis Web (ASW)**, **Disseny de Bases de Dades (DBD)**, **Intel·ligència Artificial (IA)**, **Projecte d'Enginyeria de Software (PES)**, **Conceptes Avançats de Programació (CAP)**, **Projectes de Programació (PROP)** i **Videojocs (VJ)**. Aquestes assignatures m'han donat els coneixements necessaris per desenvolupar una aplicació web fent amb una RESTful API, gestionar una base de dades SQL, implementar un model de *Machine Learning*, construir un sistema fent servir bones pràctiques de programació i aprendre noves tecnologies que mai havia fet servir.

Encara sent molts els coneixements adquirits en assignatures de carrera, la investigació i treball personal a l'hora de descobrir noves tecnologies és molt important, no només en un

projecte com aquest, sinó en qualsevol projecte que es pugui realitzar en l'àmbit professional.

15.4. Treball futur

És molt freqüent durant el desenvolupament d'un projecte, pensar possibles funcionalitats o millores que donen al producte un valor major a l'original. Algunes d'aquestes funcionalitats poden ser implementades, ja sigui perquè són una variació del que ja estava pensat o perquè no tenen un impacte que canviï el transcurs del projecte. Durant aquest apartat es definiran aquelles funcionalitats que no s'han pogut implementar, sigui per falta de temps o una modificació en el context original.

Utilitzar *hashtags* o missatges d'Instagram. Instagram és una de les xarxes socials més importants del món. Aquesta aplicació s'utilitza majoritàriament per compartir fotos o vídeos amb tot el món. Semblant a Twitter, Instagram també utilitza *hashtags* per a agrupar publicacions. Una futura implementació pot ser utilitzar l'API d'Instagram per a fer un *sentiment analysis* de *hashtags* d'Instagram o comentaris d'una publicació tal com hem fet amb Twitter.

Expansió de llenguatges. Twitter és una xarxa social global que s'utilitza en tot el món, és per això que la diversitat cultural i d'idiomes és màxima. Una futura implementació en el model de *Machine Learning* pot ser utilitzar conjunts de dades en altres idiomes per tal de que el model pugui predir el *sentiment* de qualsevol tweet sigui el llenguatge que sigui.

Monitoratge de l'aplicació. Tota aplicació ha de tenir un monitoratge constant per part de personal o *software* especialitzat i tenir una idea del rendiment que està donant. Aquesta funcionalitat és clau per a aplicacions a llarg termini que busquen estar en el mercat durant molt de temps. És per això que és important un monitoratge de l'aplicació per saber que pot fallar i poder corregir errors.

Augmentar la precisió del model. Encara que el model de *Machine Learning* desenvolupat té una precisió més que adequada, aquesta és millorable utilitzant diverses tècniques com la integració d'altres conjunts de dades dins de l'utilitzat. Per exemple, augmentant el nombre de "Neutrals" en el conjunt de dades i el percentatge de registres per l'entrenament, ajudaria a augmentar fins a un 5-6% més la precisió del model i, per tant, de les seves prediccions.

15.5. Valoració personal

Per finalitzar, aquest projecte ha sigut un dels més grans i del que més esforç i hores hi he dedicat. Ha sigut una experiència única, ja que he après sobre xarxes neuronals, que encara tenint una molt petita base darrere, sempre havia tingut especial interès en el tema i mai havia tingut l'oportunitat de desenvolupar un projecte que hem fet aprofundir en aquest tema.

Durant el transcurs del projecte hi ha hagut moments de bloqueig i altres moments on tot sortia com estava planificat. Aquests moments m'han servit per adonar-me de la importància de la constància i que, encara que al començament el projecte sigui costa amunt i sigui difícil, sempre es pot completar amb ganes, esforç i seguint una planificació preestablerta. He après a desenvolupar totes les parts necessàries per construir una aplicació, he après a implementar *Machine Learning* en una aplicació, he après a gestionar i planificar un projecte de *software* i, gràcies al suport dels meus directors, superar els contratemps i donar una solució correcta al problema plantejat inicialment.

He de dir que sempre he sigut una persona que li aterrava els començaments de treballs que, a priori, poden semblar complicats. Avui dia, puc dir que en completar aquest treball, he après a eliminar aquest pensament i veure que, de mica en mica, qualsevol cosa es pot aconseguir. I això em dona un valor incalculable per al futur, sigui per a l'àmbit laboral a l'hora de dur a terme projectes, com en l'àmbit personal en el dia a dia.

Bibliografia

- [1] **Esan**, 6 obstáculos que todo gestor de proyectos debe superar. 2019. [Online]. <https://www.esan.edu.pe/apuntes-empresariales/2019/11/6-obstaculos-que-todo-gestor-de-proyectos-debe-superar/>. [Consultat en 22/09/2021]
- [2] **J. Robertson, S. Robertson**, Volere Requirements Specification Template. 2015. [Offline]. [Consultat en 23/09/2021]
- [3] **Brand24**, Media Monitoring. 2021. [Online]. <https://brand24.com/>. [Consultat en 23/09/2021]
- [4] **Talkwalker**. 2021. [Online]. <https://www.talkwalker.com/es#>. [Consultat en 23/09/2021]
- [5] **Hashtagify**, Find and Analyze Hashtags. 2021. [Online]. <https://hashtagify.me>. [Consultat en 23/09/2021]
- [6] **Wikipedia**, Scrum (software development). 2021. [Online]. [https://en.wikipedia.org/wiki/Scrum_\(software_development\)](https://en.wikipedia.org/wiki/Scrum_(software_development)). [Consultat en 24/09/2021]
- [7] **Github**, mllint - Linter for Machine Learning. 2021. [Online]. <https://bvobart.github.io/mllint/>. [Consultat en 24/09/2021]
- [8] **Wikipedia**, Agile software development. 2021. [Online]. https://en.wikipedia.org/wiki/Agile_software_development. [Consultat en 26/09/2021]
- [9] **Wikipedia**, Sprint (Scrum). 2021. [Online]. [https://en.wikipedia.org/wiki/Scrum_\(software_development\)#Sprint](https://en.wikipedia.org/wiki/Scrum_(software_development)#Sprint). [Consultat en 26/09/2021]
- [10] **Khan Mohammad Habibullah, Jennifer Horkoff**, Non-functional Requirements for Machine Learning: Understanding Current Use and Challenges in Industry. 2021 [Online]. <https://arxiv.org/pdf/2109.00872.pdf>. [Consultat en 06/10/2021]
- [11] **Jennifer Horkoff**, Non-Functional Requirements for Machine Learning: Challenges and New Directions. 2020 [Online].

<http://www.cse.chalmers.se/~jenho/PaperFiles/NFRsforMLRENext.pdf>. [Consultat en 06/10/2021]

[12] **Lucidchart**, Diagrama de Gantt Online. 2021 [Online].
<https://www.lucidchart.com/pages/es/ejemplos/diagrama-de-gantt-online>. [Consultat en 05/10/2021]

[13] **Ofertas de Trabajo**, Bolsa de Trabajo | Buscar Empleo En Indeed España. 2021 [Online].
<https://es.indeed.com/>. [Consultat en 10/10/2021]

[14] **Glassdoor**, Sueldos de la empresa. 2021 [Online].
<https://www.glassdoor.es/Sueldos/index.htm>. [Consultat en 10/10/2021]

[15] **EUNCET Business School**, Análisis de desviaciones | ¿Para qué sirve? 2021 [Online].
<https://blog.euncet.es/analisis-de-desviaciones-para-que-sirve/>. [Consultat en 11/10/2021]

[16] **Tarifasgasluz.com**, Precio del kWh hoy en el mercado regulado. 2021 [Online].
<https://tarifasgasluz.com/comparador/precio-kwh>. [Consultat en 11/10/2021]

[17] **DieseloGasolina.com**, Calcular gasto en gasolina y consumo en un viaje en coche. 2021 [Online].
<https://www.dieselogasolina.com/calculadora-de-trayecto-y-coste-de-viaje-en-coche.html>. [Consultat en 10/10/2021]

[18] **Microsoft Azure**, Precios de máquinas virtuales Windows. 2021. [Online].
<https://azure.microsoft.com/es-es/pricing/details/virtual-machines/windows/>. [Consultat en 07/10/2021]

[19] **Microsoft**, Microsoft 365. 2021 [Online]
<https://www.microsoft.com/es-es/microsoft-365/buy/compare-all-microsoft-365-products>. [Consultat en 07/10/2021]

[20] **Go, A., Bhayani, R. and Huang, L.**, 2009. Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford, 1(2009), p.12.

[21] **Wikipedia**, Twitter. 2021 [Online] <https://es.wikipedia.org/wiki/Twitter> [Consultat en 22/10/2021]

[22] **Wikipedia**, Hashtag. 2021 [Online]. <https://es.wikipedia.org/wiki/Hashtag#Origen> [Consultat en 22/10/2021]

[23] **Jennifer Horkoff**, Non-Functional Requirements for Machine Learning: Challenges and New Directions. 2021 [Online] <http://www.cse.chalmers.se/~jenho/PaperFiles/NFRsforMLRENext.pdf> [Consultat en 25/10/2021]

[24] **TensorFlow**, Text classification. 2021 [Online] https://www.tensorflow.org/lite/examples/text_classification/overview [Consultat en 02/11/2021]

[25] **Universidad de Pamplona**, Buenas prácticas aplicadas a la implementación colaborativa de aplicativos web. [Consultat en 04/11/2021]

[26] **NubeColectiva**, 5 Buenas Prácticas de Machine Learning para Desarrolladores en Python. 2020 [Online] <https://blog.nubecolectiva.com/5-buenas-practicas-de-machine-learning-para-desarrolladores-en-python/> [Consultat en 04/11/2021]

[27] **NetApp**, ¿Qué es DevOps?. 2020 [Online] <https://www.netapp.com/es/devops-solutions/what-is-devops/> [Consultat en 04/11/2021]

[28] **Radu Dimitriu Boboia**, Herramientas y buenas prácticas para el desarrollo, mantenimiento y evolución de Software en Java. 2019 [Online] https://oa.upm.es/55663/1/TFG_RADU_DUMITRU_BOBOIA.pdf [Consultat en 04/11/2021]

[29] **Wikipedia**, Metodología de desarrollo de software. 2021 [Online] https://es.wikipedia.org/wiki/Metodolog%C3%ADa_de_desarrollo_de_software [Consultat en 06/11/2021]

[30] **RedHat**, ¿Qué son la integración y la distribución continuas (CI/CD)? 2018 [Online] <https://www.redhat.com/es/topics/devops/what-is-ci-cd> [Consultat en 06/11/2021]

[31] **TensorFlow**. 2021 [Online] <https://www.tensorflow.org/?hl=es-419> [Consultat en 06/11/2021]

[32] **SE-ML**. Ensure Data Labelling is Performed in a Strictly Controlled Process. 2021 [Online] https://se-ml.github.io/best_practices/01-data-label/ [Consultat en 06/11/2021]

[33] **Silverio Martínez-Fernández, Justus Bogner, Xavier Franch, Marc Oriol, Julien Siebert, Adam Trendowicz, Anna Maria Vollmer, Stefan Wagner**, Software Engineering for AI-Based Systems: A Survey. 2021 [Online] <https://arxiv.org/pdf/2105.01984.pdf> [Consultat en 10/11/2021]

[33] **GitHub**, Python Development Workflow for Humans. 2021 [Online]. <https://github.com/pypa/pipenv> [Consultat en 15/11/2021]

[33] **Jeremy Jordan**, Effective testing for machine learning systems. 2020 [Online] <https://www.jeremyjordan.me/testing-ml/> [Consultat en 15/11/2021]

[34] **Goku Mohandas**, Testing ML Systems: Code, Data and Models. 2021 [Online] <https://madewithml.com/courses/mlops/testing/> [Consultat en 15/11/2021]

[35] **Pylint**, Running Pylint. 2021 [Online] https://pylint.pycqa.org/en/latest/user_guide/run.html [Consultat en 17/11/2021]

[36] **GitHub**, A curated list of pylint errors. 2021. [Online] <https://vald-phoenix.github.io/pylint-errors/> [Consultat en 17/11/2021]

[37] **Diagrams.net**, Flowchart Maker Online. 2021. [Online] <https://app.diagrams.net/> [Consultat en 02/12/2021]

[38] **Oracle**, ¿Qué es la inteligencia artificial—IA?. 2021 [Online] <https://www.oracle.com/mx/artificial-intelligence/what-is-ai/> [Consultat en 03/12/2021]

[39] **Wikipedia**, Inteligencia artificial. 2021 [Online] https://es.wikipedia.org/wiki/Inteligencia_artificial#Historia [Consultat en 03/12/2021]

[40] **Wikipedia**, Prueba de Turing. 2021 [Online] https://es.wikipedia.org/wiki/Prueba_de_Turing [Consultat en 03/12/2021]

[41] **Rodrigo Alonso**, IA, Machine Learning y Deep Learning, ¿cuál es la diferencia? 2021 [Online] <https://hardzone.es/tutoriales/rendimiento/diferencias-ia-deep-machine-learning/> [Consultat en 03/12/2021]

[42] **BBVA**, Machine learning: ¿qué es y cómo funciona? 2021 [Online] <https://www.bbva.com/es/machine-learning-que-es-y-como-funciona/> [Consultat en 03/12/2021]

[43] **Viquipèdia**, Escacs per ordinador. 2021 [Online] https://ca.wikipedia.org/wiki/Escacs_per_ordinador [Consultat en 03/12/2021]

[44] **Jordi Torres**, Redes Neuronales Recurrentes. 2019 [Online] <https://torres.ai/redes-neuronales-recurrentes/> [Consultat en 04/12/2021]

[45] **Wikipedia**, Long short-term memory. 2021 [Online] https://en.wikipedia.org/wiki/Long_short-term_memory [Consultat en 04/12/2021]

[46] **Afshine Amidi, Shervine Amidi**, Recurrent Neural Networks cheatsheet. 2021 [Online] <https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-recurrent-neural-networks>
<https://arxiv.org/pdf/2103.07286.pdf> [Consultat en 04/12/2021]

[47] **Silverio Martínez-Fernández**, AI engineering. How to engineer AI-based systems? 2021 [Online] <https://zonavideo.upc.edu/video/615ee3e702108030ff013998> [Consultat en 08/12/2021]

[48] **Wikipedia**, Natural language processing. 2021 [Online] https://en.wikipedia.org/wiki/Natural_language_processing [Consultat en 08/12/2021]

[49] **Aravindpai Pai**, What is Tokenization in NLP? Here's All You Need To Know. 2020 [Online] <https://www.analyticsvidhya.com/blog/2020/05/what-is-tokenization-nlp/> [Consultat en 08/12/2021]

[50] **Jason Brownlee**, What is a Confusion Matrix in Machine Learning. 2020 [Online] <https://machinelearningmastery.com/confusion-matrix-machine-learning/> [Consultat en 04/01/2022]

[51] **Palash Sharma**, Keras Dense Layer Explained for Beginners. 2020 [Online] <https://machinelearningknowledge.ai/keras-dense-layer-explained-for-beginners/> [Consultat en 04/01/2022]

[52] **Sawan Saxena**, Understanding Embedding Layer in Keras. 2020 [Online] <https://medium.com/analytics-vidhya/understanding-embedding-layer-in-keras-bbe3ff1327ce> [Consultat en 04/01/2022]

[53] **Keras**, Dropout Layer. 2020 [Online] https://keras.io/api/layers/regularization_layers/dropout/ [Consultat en 04/01/2022]

[54] **Jason Brownlee**, Overfitting and Underfitting With Machine Learning Algorithms. 2019 [Online] <https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/>

[54] **Bart van Oort**, Code Smells & Software Quality in Machine Learning Projects. 2021 [Online] <https://repository.tudelft.nl/islandora/object/uuid:b20883f8-a921-487a-8a65-89374a1f3867?collection=education> [Consultat en 05/01/2022]

[55] **Apoorv Agarwal, Boyi Xie, Ilia Vosha, Owen Rambow i Rebecca Passonneau**, Sentiment Analysis of Twitter Data. 2011 [Online] <https://aclanthology.org/W11-0705.pdf> [Consultat en 06/01/2022]

[56] **Ariadna de Arriba, Marc Oriol i Xavier Franch**, Applying Transfer Learning to Sentiment Analysis in Social Media. 2021 IEEE 29th International Requirements Engineering Conference Workshops (REW) [Online] <https://ieeexplore.ieee.org/document/9582287> [Consultat en 10/01/2022]

[57] **Ariadna de Arriba, Marc Oriol i Xavier Franch**, Applying Sentiment Analysis on Spanish Tweets using BETO. 2021 [Online] http://ceur-ws.org/Vol-2943/emoeval_paper9.pdf [Consultat en 10/01/2022]

[58] **Rafa Arjonilla**, ¿Qué es el Backend?. 2022 [Online] <https://rafarjonilla.com/que-es/backend/> [Consultat en 12/01/2022]

Annex

A. Manual d'usuari

A continuació es mostraran les diferents pantalles que constitueixen l'aplicació i com utilitzar-les per tal de treure el major rendiment possible al servei dispostat per un usuari que mai hagi utilitzat la pàgina web anteriorment.

A.A. Pantalla principal

La primera pantalla que veu l'usuari és la de cerca d'un hashtag. Com hem mencionat anteriorment, un usuari pot cercar hashtags per veure la seva anàlisi sense necessitat d'estar registrat o en una sessió dins de l'aplicació. Aquesta pantalla disposa d'un cercador i dos filtres, un pel nombre de tweets que s'agafen per fer l'anàlisi i un altre per introduir la data fins a la qual s'agafen els tweets, si no s'introdueix res en cap dels dos filtres, el nombre de tweets que s'agafen serà 100. En el cercador es pot introduir un *hashtag* o una paraula clau qualsevol. Per realitzar l'anàlisi l'usuari haurà de pressionar el botó de "Submit". No només això sinó que en aquesta pantalla l'usuari, sempre que estigui loguejat, pot veure les sol·licituds d'amistat i afegir amics de forma ràpida.

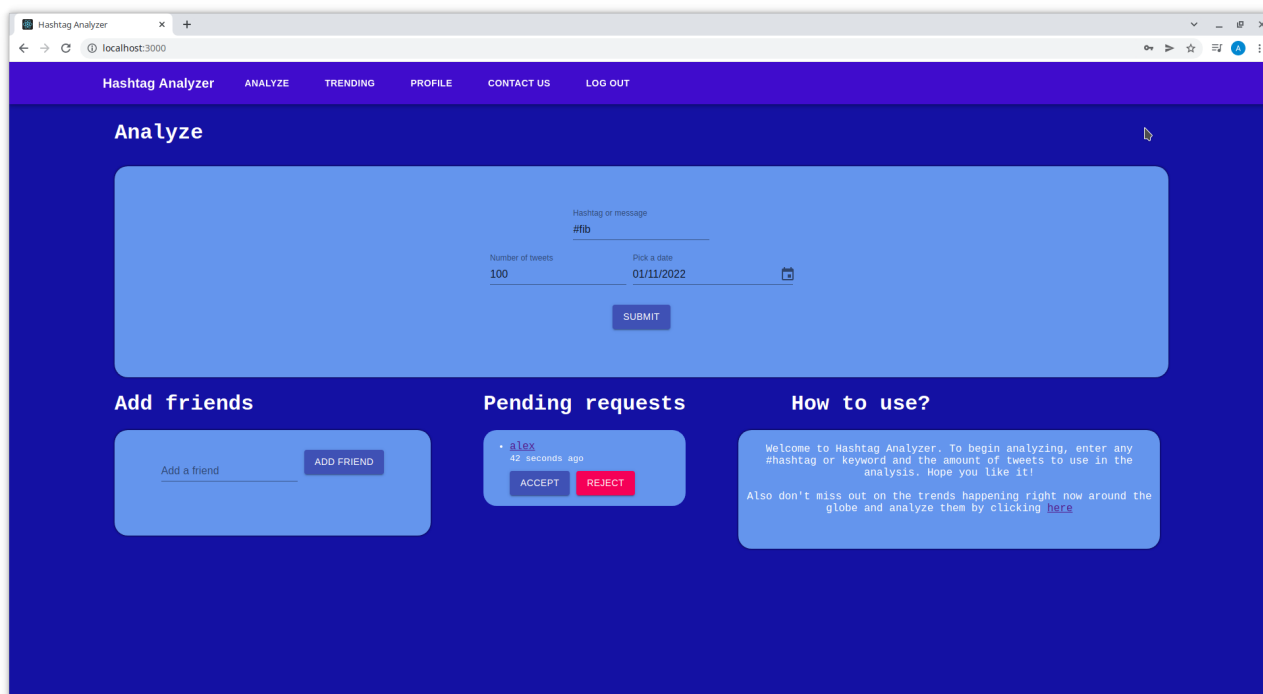


Figura 25: Pantalla principal

A.B. Pantalla d'anàlisi de hashtags

Aquesta pantalla mostra la funcionalitat principal de l'aplicació que és la de l'anàlisi d'un hashtag i s'accedeix una vegada es prem el botó de *submit* a la pantalla de cerca de *hashtags*. En aquesta pantalla es mostra una *piechart* que dona informació sobre el percentatge de tweets positius, negatius i neutrals, aquest gràfic és interactiu, és a dir, al prémer una de les seccions només es mostraran per pantalla els tweets amb la secció seleccionada. També es mostren els tweets amb més interaccions, top tweets, i en prémer sobre un d'ells es redigirà directament al tweet original de la pàgina de Twitter. A més a més es mostraran els tweets que s'han utilitzat per tal de realitzar l'anàlisi, amb un identificador de si el seu sentiment és positiu, negatiu o neutral, juntament un botó per notificar si la predicció del sentiment ha sigut equivocada. No només això, sinó que un usuari loguejat pot guardar aquesta anàlisi en el seu perfil clicant el botó de *Save Analysis*.

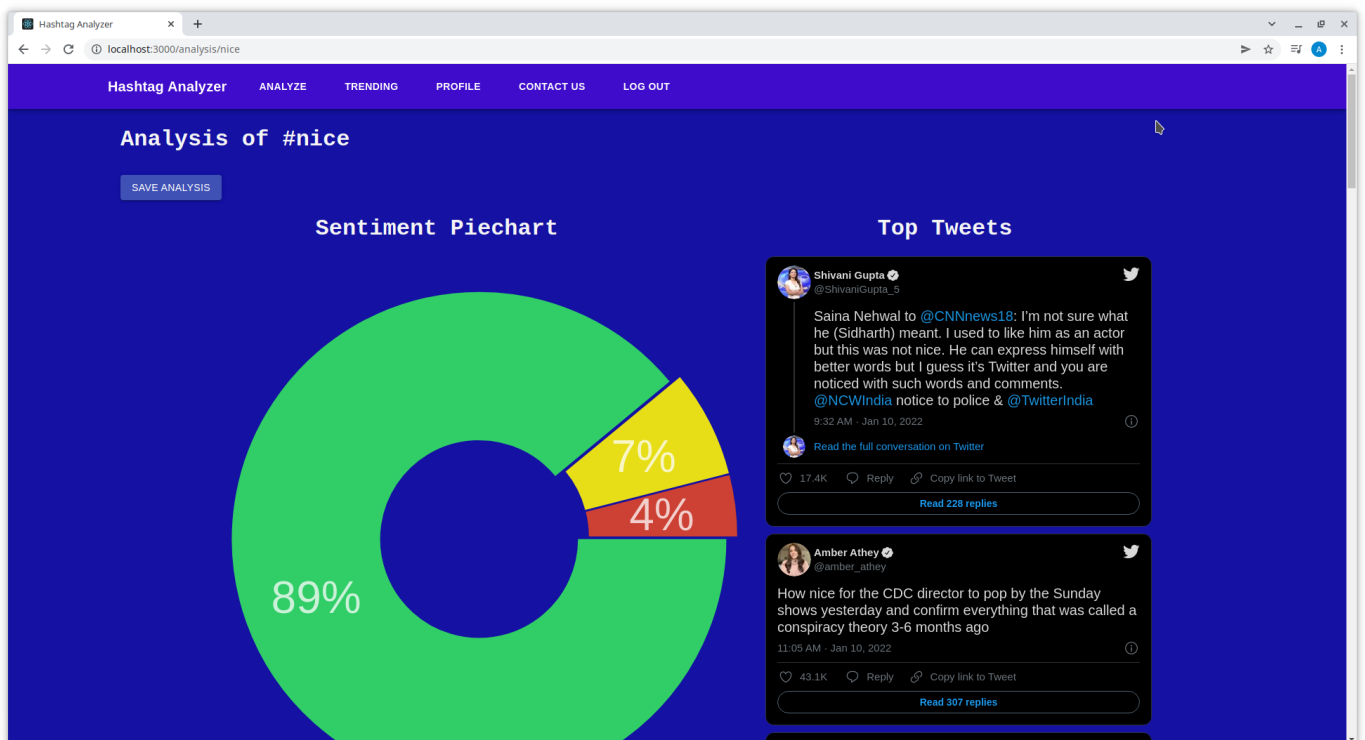


Figura 26: Pantalla d'anàlisi d'un hashtag

A.C. Pantalla d'inici de sessió

Aquesta pantalla es pot accedir en tot moment al pressionar l'enllaç de *Login* en la barra de navegació que hi ha en la part superior de totes les pantalles. En aquesta pantalla hi haurà dos camps, un per a introduir el nom d'usuari i un altre per introduir la contrasenya. Una vegada introduïts, en prémer el botó de *Login*, si l'usuari està registrat en el sistema es carrega la pantalla de cerca de *hashtag* amb l'usuari loguejat. En aquesta pantalla també hi haurà disponibles dos botons més, un per fer *login* amb un compte de *Google* i un altre per registrar-se en el sistema en cas de no tenir un compte registrat.

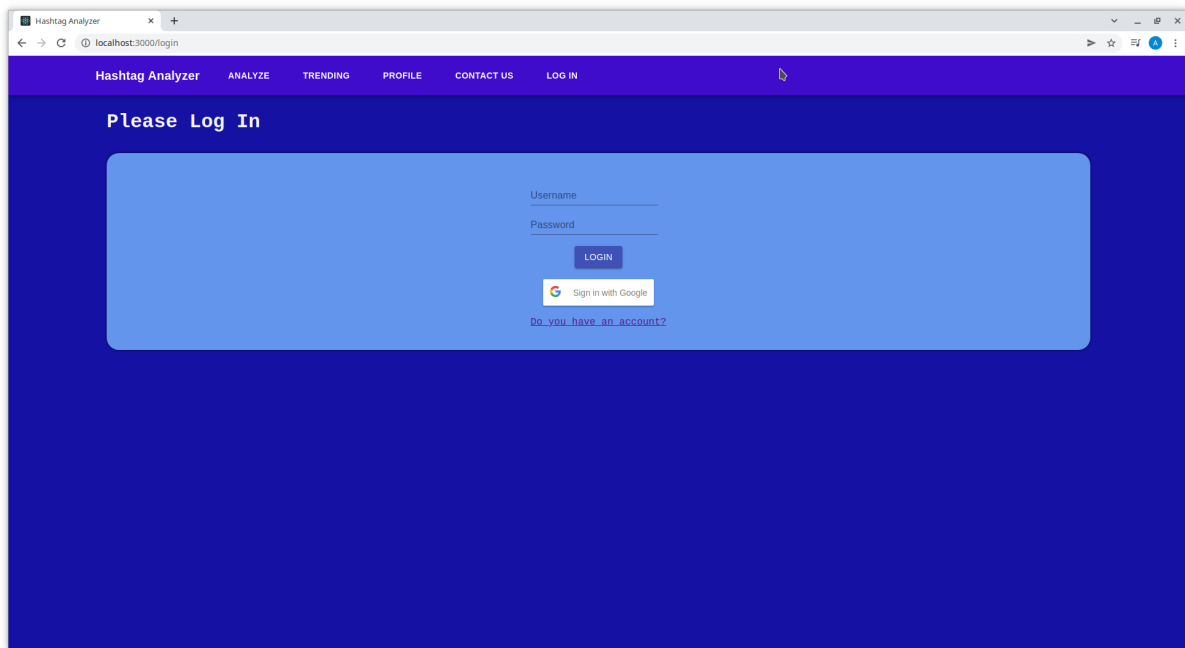
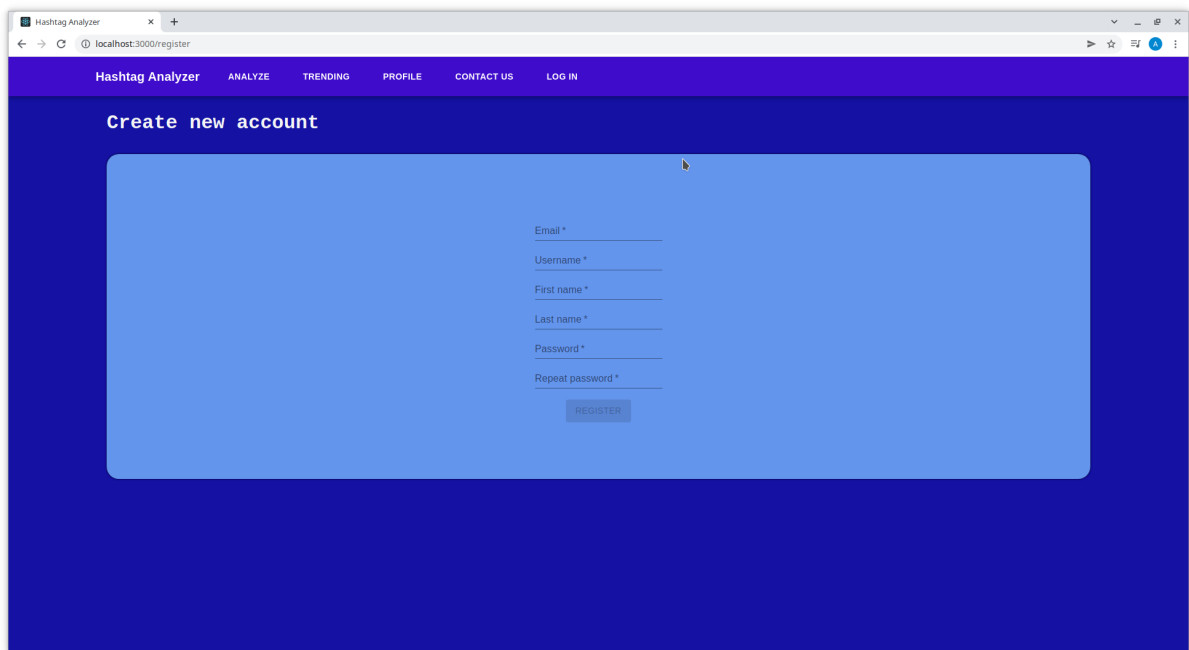


Figura 27: Pantalla d'inici de sessió

A.D. Pantalla de registre de compte

Aquesta pantalla s'accedeix en pressionar el botó de registre en la pantalla d'inici de sessió. En aquesta pantalla l'usuari introdueix l'email que es vol utilitzar, el nom d'usuari, la contrasenya dues vegades i el nom i cognom de l'usuari. En cas que el nom d'usuari o email ja existeix en el sistema es mostra un error descriptiu. També, en cas que la contrasenya no sigui de mínim 8 o que no coincideixen es mostrarà un error. Una vegada l'usuari ha completat els camps amb els requisits correctament, al pressionar el botó de registre es redigireix a la pantalla de cerca de *hashtag* amb l'usuari loguejat i registrat a l'aplicació.



The screenshot shows a web browser window titled 'Hashtag Analyzer' with the URL 'localhost:3000/register'. The page features a dark blue navigation bar with the following links: 'Hashtag Analyzer', 'ANALYZE', 'TRENDING', 'PROFILE', 'CONTACT US', and 'LOG IN'. Below the navigation bar, the heading 'Create new account' is displayed. The registration form is contained within a light blue rectangular area and includes the following input fields, each marked with an asterisk to indicate it is required: 'Email *', 'Username *', 'First name *', 'Last name *', 'Password *', and 'Repeat password *'. A 'REGISTER' button is positioned at the bottom of the form.

Figura 28: Pantalla d'inici de registre de compte

A.E. Pantalla de perfil d'usuari

Aquesta pantalla s'accedeix clicant el link de *My Profile* que està disponible en la barra superior que és present en totes les pantalles. En aquesta pantalla un usuari pot veure les seves dades personals, ja sigui l'email, nom d'usuari, data de registre i *about me*. Aquest últim camp és modificable i es pot guardar prement el botó de *Update about*. També es mostraran les anàlisis realitzades que ha decidit guardar per tal poder veure'ls una altra vegada o compartir-los. També pot canviar la contrasenya existent sempre que compleixi els requisits necessaris clicant el botó de *Change password*. No només això sinó que pot observar els seus amics i clicar sobre un d'ells per tal d'accedir a aquest mateix perfil però de l'amic seleccionat.

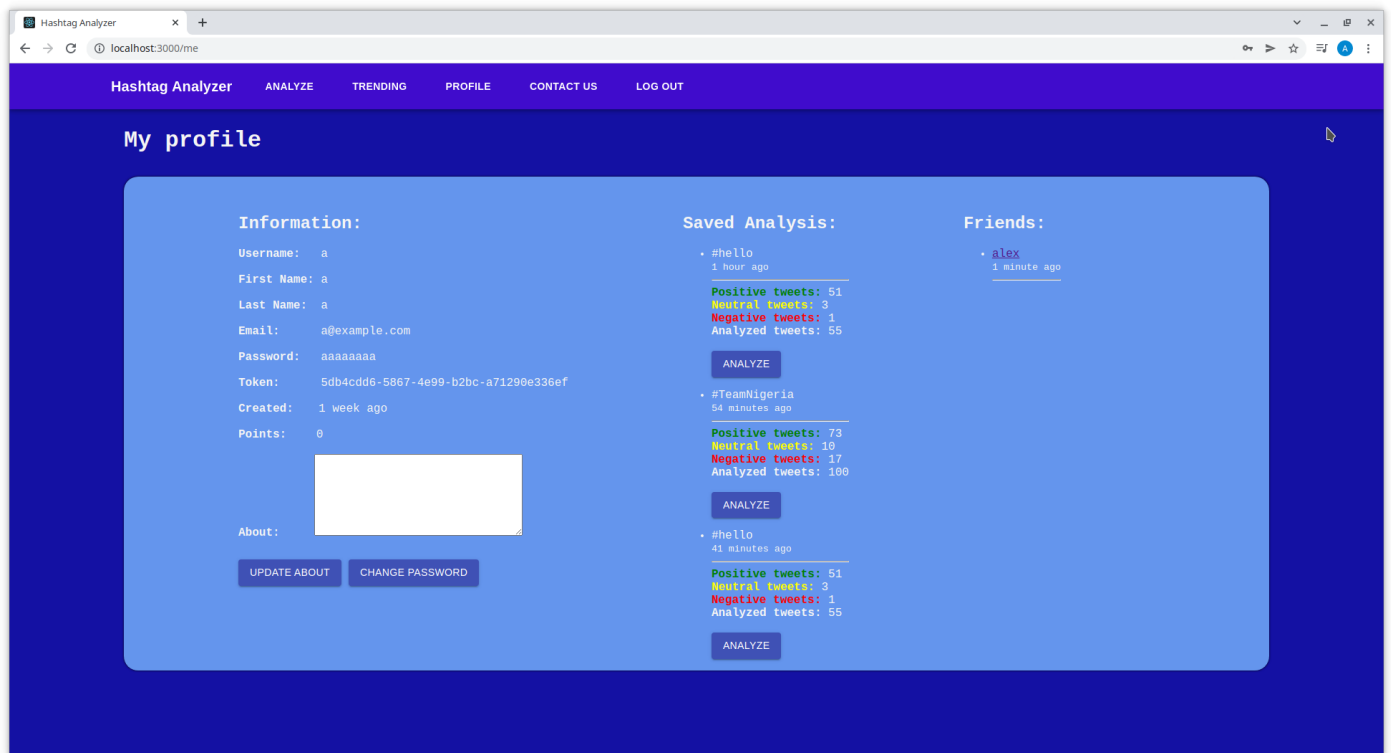


Figura 29: Pantalla de perfil d'usuari

A.F. Pantalla de trending

Aquesta pantalla s'accedeix mitjançant el link de *Trending* en la barra de *links* en la part superior de totes les pantalles. Es mostren, inicialment, els 10 hashtags o paraules claus *trending* arreu del món. Es disposarà d'un filtre per tal d'escollir un país i es mostren els *trending* del país corresponent. Tot *hashtag* o paraula clau té un botó al costat per tal de realitzar l'anàlisi d'aquest/a directament. A més a més els *hashtags* disposen d'un botó per tal de poder publicar un tweet amb aquell hashtag directament en la pàgina oficial de Twitter.

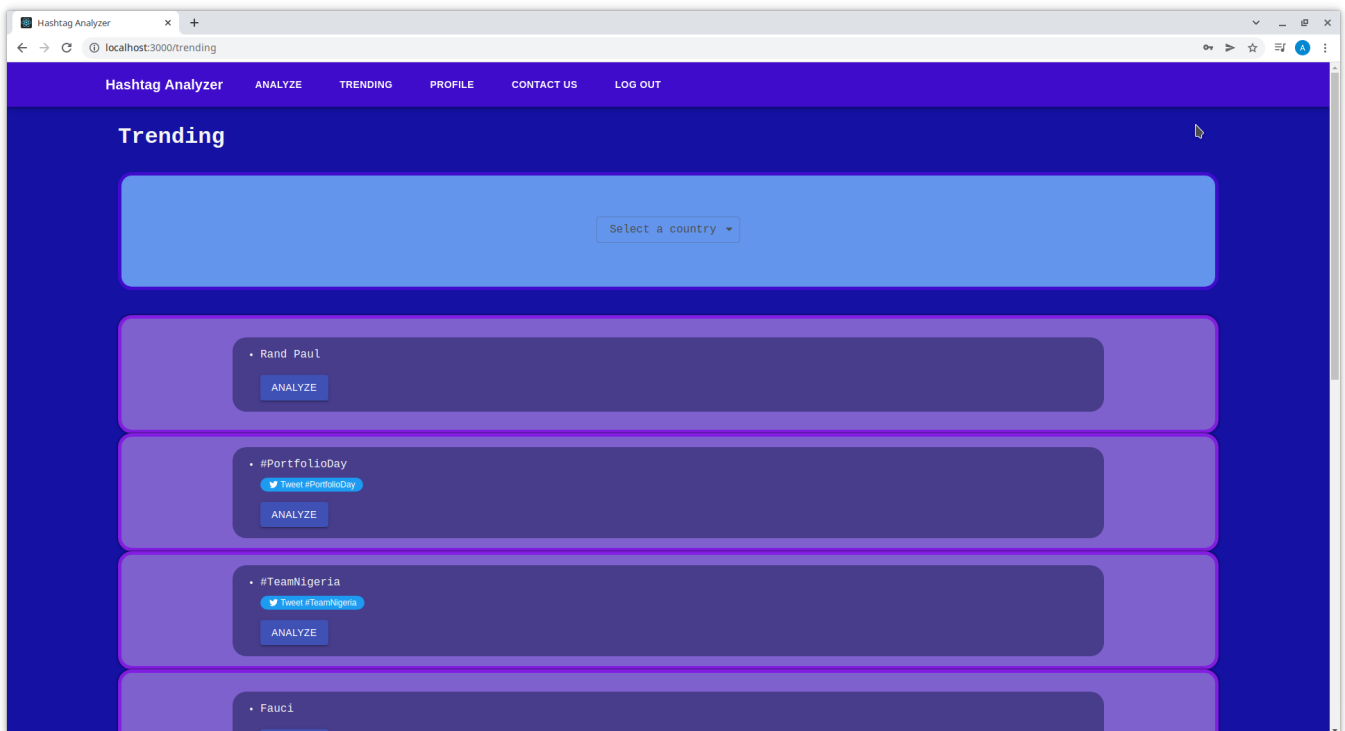


Figura 30: Pantalla de trending

A.G. Pantalla de contacte

Aquesta pantalla és accessible amb el link de *Contact Us* en la barra superior de links. En aquesta pantalla es mostra informació sobre l'aplicació, l'autor i requisits legals que s'han de tenir en compte per utilitzar aquesta aplicació.



Figura 31: Pantalla de contacte