



Social Network monitoring

Grau Enginyeria Informàtica – Especialitat Enginyeria Software

Alumne: Joan Colom Montoya

Director: Jordi Marco Gómez

Codirector: Marc Oriol Hilari

Universitat Politècnica de Catalunya

Abstracte

Facultat d'Informàtica de Barcelona Enginyeria de Serveis i Sistemes de
la Informació

Grau d'Enginyeria Informàtica

Social Network monitoring

per Joan COLOM MONTOYA

El monitoratge consisteix en comprovar de manera continua l'estat d'un sistema per tal d'assegurar-ne la fiabilitat, la seguretat o, simplement, donar aquesta informació a l'usuari. El monitoratge és un procés que requereix de bastants recursos, tant software com hardware. En funció de la informació que rep l'usuari que utilitza el sistema de monitoratge, aquest pot decidir fer un acció o altra. Partint del fet de monitoritzar diverses xarxes socials, com es pot utilitzar una arquitectura de software eficient que permeti fer el monitoratge en diverses xarxes socials?

En base a aquesta premissa, aquest TFG intenta donar resposta a aquest fet.

Universitat Politècnica de Catalunya

Abstract

Facultat d'Informàtica de Barcelona Enginyeria de Serveis i Sistemes de
la Informació

Grau d'Enginyeria Informàtica

Social Network monitoring

by Joan COLOM MONTOYA

Monitoring consists on continuously checking the status of a system due to ensure the reliability, security or, simply, giving information to the user. This is a process that requires many resources, both software and hardware. Depending on the information received by the system user, the user may decide one or another course of action. Assuming the fact of monitoring various social networks, how could an efficient software architecture be used and allow these social networks monitoring?

Based on this premise, this TFG tries to give a response to this fact.

Agraïments

Vull agrair a tothom que ha fet possible la realització del TFG:

A en Jordi Marco, el director del projecte, que ha donat un enfocament global al treball.

A en Marc Oriol, el codirector del projecte, que sempre ha estat per donar-me un cop de mà i per mostrar-me el camí a seguir per poder desenvolupar el projecte de forma autònoma, com poder investigar i com poder sortir-ne quan he estat parat sense saber quin camí seguir.

Juntament, al director i al codirector del projecte, que han tingut paciència amb mi per a que pugui avançar al meu ritme i aturar temporalment el treball per temes de salut.

També a la meva mare i al meu pare, que sempre m'han animat a fer el projecte i no deixar-lo de banda començant a treballar tenint-lo a mitges, mostrant-me que si el deixo pendent, sempre quedarà així.

Per finalitzar, donar-li les gràcies a la meva germana i al seu marit per animar-me a continuar i ensenyar-me amb el seu exemple la positivitat de seguir endavant.

Índex

1	Introducció	19
1.1	<i>Motivació del projecte</i>	19
1.2	<i>Àrea d'estudi i justificació de la temàtica</i>	19
1.3	<i>Identificació dels stakeholders</i>	20
2	Contextualització i estat de l'art	22
2.1	<i>Context</i>	22
2.1.1	Introducció	22
2.1.2	Director del projecte	23
2.1.3	Codirector del projecte	23
2.1.4	Desenvolupador	23
2.1.5	Usuaris.....	23
2.2	<i>Estat de l'art</i>	24
2.2.1	Contextualització.....	24
2.2.2	Estudi de mercat	26
2.2.2.1	Hotsuite [9]	26
2.2.2.2	Brandwatch [10]	26
2.2.2.3	Media monitoring API de mention [11].....	26
2.2.3	Conclusions de l'estudi de mercat	27
3	Formulació del problema i abast	29
3.1	<i>Objectius</i>	29
3.1.1	Objectius principals	29
3.1.2	Objectius específics	29
3.2	<i>Abast</i>	29
3.3	<i>Possibles obstacles</i>	30
3.3.1	Desconeixement de la tecnologia	30
3.3.2	Impediments tecnològics causats per la API externa	30
3.3.3	Manca de temps.....	31
3.3.4	Catàstrofes tecnològiques.....	31
3.3.5	Problemes de salut.....	31
4	Metodologia i rigor	33
4.1	<i>Introducció a KANBAN</i>	33
4.2	<i>Explicació de la metodologia KANBAN</i>	34
4.3	<i>Iteracions setmanals o cada dues setmanes</i>	34

4.4	<i>Reunions (semi)presencials setmanals o cada dues setmanes</i>	35
4.5	<i>Eines de seguiment</i>	35
4.6	<i>Mètode de validació</i>	35
5	Planificació temporal	37
5.1	<i>Estimació temporal del projecte</i>	37
5.2	<i>Recursos</i>	37
5.2.1	Recursos personals.....	37
5.2.2	Recursos materials	37
5.3	<i>Pla d'acció i alternatives</i>	38
5.4	<i>Calendari</i>	39
5.4.1	Estimació d'hores i dates	39
5.4.2	Dates importants al calendari	40
5.4.3	Descripció de les tasques	40
5.4.4	Dependències entre tasques.....	42
5.4.5	Diagrama de Gantt	43
6	Gestió econòmica	49
6.1	<i>Consideracions a tenir en compte</i>	49
6.2	<i>Identificació i estimació dels costos</i>	49
6.2.1	Costos directes per activitat.....	51
6.2.2	Costos indirectes	52
6.2.3	Contingència.....	54
6.2.4	Imprevistos	54
6.2.5	Pressupost	55
7	Sostenibilitat i viabilitat del projecte	57
7.1	<i>Econòmica</i>	57
7.2	<i>Social</i>	57
7.3	<i>Ambiental</i>	58
8	Sistema de monitoratge	60
8.1	<i>Monitors</i>	60
8.1.1	Especificacions tècniques i funcionals	61
8.1.1.1	Comportament intern del monitor	62
8.1.1.2	Ús de les dades obtingudes pel monitor	66
8.1.1.3	Configuració del monitor	67
8.1.2	Disseny i arquitectura genèrica.....	69
8.1.3	Implementació de monitors	72

8.2	<i>Tecnologies utilitzades</i>	75
9	Interacció amb el sistema	78
9.1	<i>Anàlisi de requisits</i>	78
9.2	<i>Disseny de la interfície</i>	78
9.3	<i>Implementació</i>	81
10	Validació del sistema	83
10.1	<i>Disseny de tests</i>	83
10.2	<i>Execució de tests</i>	83
11	Conclusions	85
11.1	<i>Avaluació del potencial del producte generat</i>	85
11.2	<i>Avaluació general</i>	85
12	Referències	88

Índex de figures

Fig 1 Codi Qr amb l'enllaç al diagrama de Gantt de GEP	44
Fig 2 Diagrama de Gantt de GEP	45
Fig 3 Codi Qr amb l'enllaç al diagrama de Gantt de la resta del TFG	46
Fig 4 Diagrama de Gantt de la resta del TFG	47
Fig 5 Exemple JSON del recurs “comentaris” del Simulador	64
Fig 6 Exemple JSON del recurs “mencions” del Simulador	65
Fig 7 JSON amb les dades d'un element de la monitorització.....	66
Fig 8 Ús de les dades.....	67
Fig 9 Diagrama de les classes Java dels monitors.....	69
Fig 10 Estructura interfície gràfica.....	78

Índex de taules

Taula 1 Estimació d'hores i dates	39
Taula 2 Cost total per rol	49
Taula 3 Costos directes per activitat	51
Taula 4 Costos indirectes	53
Taula 5 Costos de la contingència	54
Taula 6 Costos dels imprevistos	54
Taula 7 Cost total	55

Glossari

En aquest apartat, definirem la terminologia tècnica i específica que es fa servir al llarg del document i projecte.

- **API (*Application Programming Interface*):** és una interfície que indica com diferents programes informàtics haurien d'interactuar.
- **IA (*Intel·ligència Artificial*):** és un concepte relacionat amb la informàtica, vinculat al fet que les màquines puguin prendre decisions semblants (o millors) a les que prendria un ésser viu. Aquestes decisions es prenen tenint en compte la percepció, raonament, aprenentatge, comunicació i actuar en entorns complexos [7].
- **Monitor:** eina que s'encarrega de buscar tots els elements generats pel monitor, que es poden filtrar per una paraula clau, amb una determinada freqüència de temps i que pertanyen a la xarxa social d'Instagram. El monitor pot realitzar la seva funció gràcies a l'API que li permet fer-ho (Instagram Graph API).
- **Open source:** quelcom que les persones poden modificar perquè és públicament accessible .
- **Sentiment Analysis:** a l'àmbit d'aquest TFG fa referència a l'extracció de dades a partir dels textos que els diferents usuaris han publicat a les xarxes socials, amb l'objectiu de saber què pensen o quines són les seves emocions respecte un tema determinat. Tot això esmentat, tenint les dades de forma objectiva segons el criteri fixat per la persona que fa l'anàlisi de les dades.
- **Servei web:** tecnologia que usa un conjunt de protocols i estàndards que serveixen per a intercanviar dades entre aplicacions.
- **Software open source:** és el software amb el codi font que qualsevol pot inspeccionar, modificar i ampliar / millorar.
- **TimeSlot:** freqüència de temps amb la que volem generar dades monitoritzades d'una consulta específica.
- **Xarxa social:** aplicació web que serveix per compartir contingut amb els teus contactes o resta d'usuaris de la xarxa social en qüestió. També en el sentit contrari, per consumir contingut generat per altres. Alguns exemples de xarxes socials són: Facebook, Instagram, Twitter, YouTube, ...

1 Introducció

1.1 Motivació del projecte

Aquest projecte té diverses motivacions.

Per començar, és un Treball Final de Grau (TFG), per tant, hi ha diverses motivacions implícites dins de tot TFG:

- Obtenir el títol que acredita els estudis realitzats. Aquest és necessari per poder realitzar diferents feines al sector públic.
- Aprendre a realitzar un treball a partir de tots els coneixements i conceptes apresos durant els estudis.
- Aprendre a investigar a partir de la base dels coneixements propis, tenint com a meta ser autodidacte i aplicar l'enginyeria necessària per la feina de tot enginyer.

Seguidament, el fet de realitzar un projecte d'aquesta magnitud, té un efecte motivador i reconfortant per a un mateix.

Per acabar, aclarir que el punt anterior té un motiu concret a nivell personal. Doncs he fet de becari en empreses on els responsables no tenen la paciència o recursos suficients per la formació d'un estudiant. Així doncs, moltes de les motivacions implícites que hi ha dins d'un TFG i que s'haurien d'obtenir amb la realització d'un programa de pràctiques a nivell universitari, a vegades no és possible. Amb la realització del TFG, sí que s'obtenen, sempre i quan es superi exitosament.

1.2 Àrea d'estudi i justificació de la temàtica

Les dades són molt importants i apreciades per nombroses entitats. Darrerament, només cal cercar una mica d'informació al respecte i ens n'adonarem que alguns sistemes operatius mòbils i xarxes socials (ambdós gratuïts), fan el negoci a partir de les dades que l'usuari aporta. Aquest producte (nosaltres), és venut a diverses empreses o organitzacions que en fan ús. Per exemple, fer perfils comercials per vendre'ns productes o, un altre exemple, fer que votem a un candidat determinat que es presenta a les eleccions.

Tot i això, no podem oblidar que aquest projecte no té cap ànim de lucre i que l'objectiu és purament acadèmic i de recerca. No es fa cap estudi de les dades ni "*sentiment analysis*".

1.3 Identificació dels stakeholders

A continuació, llistarem els potencials *stakeholders* i justificarem la nostra resposta.

- **Desenvolupadors i enginyers de software:** en aquest cas concret, considerem que aquest col·lectiu és una part interessada des de diverses vessants. Per una part, i degut a que aquest projecte deriva d'un projecte anterior, el qual és ampliat, tenim que uns dels actors interessats són els desenvolupadors i enginyers de software anteriors, doncs el seu treball té continuïtat. Sempre és d'agrair que un fet similar succeeixi després d'haver realitzat un esforç significatiu a la realització d'un projecte i les diferents tasques que el mateix comporta.

Per altra banda, tenim els desenvolupadors i enginyers de software que estan realitzant el projecte actualment. Es produeix una actualització i s'aprèn a agafar un projecte ja existent i a treballar a partir d'aquest. Es tracta, doncs, d'un cas pràctic del que ens trobarem durant la vida laboral. A la majoria d'ocasions, la nostra feina consisteix a fer això, els projectes ja existents necessiten ser mantinguts, adaptats i ampliat. Conseqüentment a la primera vessant, òbviament, també es vol que aquest projecte no acabi aquí, si no, que s'ampliï i serveixi per a un futur.

- **Gestors de projectes i experts en Sistemes d'Informació:** degut a la informació generada pel producte creat a partir del projecte actual, la fiabilitat, rigor seguit, modularitat i adaptabilitat del mateix, aquest projecte podrà ser aprofitat per gestors i experts en Sistemes d'Informació.
- **Usuaris finals:** identifiquem dos tipus d'usuaris finals, els que utilitzaran l'eina i els que seran una font d'obtenció de dades de l'eina (el seu contingut a les xarxes socials serà monitoritzat). Els usuaris finals hauran de ser empreses, entitats o altres usuaris que tinguin compte d'usuari a Instagram i Facebook de tipus negoci (*business*).

A mode de resum, concloem que aquest projecte té qualitats que el fan reaprofitable i modificable, tal com es fa patent als punts anteriors, per l'origen i la continuïtat que en aquests moments està tenint.

2 Contextualització i estat de l'art

2.1 Context

2.1.1 Introducció

Aquest projecte és un Treball Final de Grau (TFG) de la Facultat d'Informàtica de Barcelona (FIB). Parteix d'un dels projectes disponibles per a estudiants que hi ha a la pàgina web del GESSI ^{1,2}.

Les funcionalitats que es desitgen per aquest projecte són:

- Monitoritzar els comentaris que es fan a la pròpia compte d'Instagram i les mencions que se'ns fan a les nostres publicacions.
- Poder filtrar les dades monitoritzades per paraula, tal que els resultats monitoritzats hauran de satisfer el fet de contenir la paraula clau en el seu text.
- Tenir un projecte fàcil de mantenir i ampliar per tal de poder obtenir més metadades rellevants del que busquem (localització, enllaços a les fotografies d'instagram, etc.).

A partir dels resultats, es podria fer un anàlisi del sentiment, manualment, per qualsevol persona. El problema radica en el fet que hi ha molt contingut a la xarxa, inclús si filtrem per una empresa específica (o quelcom del que volem informació), quant més gran o presència tingui a la xarxa, més difícil i costós en temps esdevindrà.

Amb el sistema de monitorització desenvolupat en aquest projecte, s'automatitza aquesta tasca i es dóna una interfície agradable basada en web. Així, es podrà visualitzar la informació requerida i la seva evolució al llarg del temps.

Es donarà suport a que l'usuari descarregui / importi les dades de les seves monitoritzacions amb format JavaScript Object Notation (JSON). Doncs les dades de la monitorització en curs s'emmagatzemen a dins del propi servidor dins d'un fitxer de text que conté el JSON amb tots els resultats.

¹ <https://gessi.upc.edu/>

² <https://gessi.upc.edu/en/students/pizza-day/propostestfgs.pdf>

2.1.2 Director del projecte

És en Jordi Marco, professor agregat del departament de Computer Science (CS) de la FIB.

2.1.3 Codirector del projecte

En Marc Oriol Hilari, Personal de Suport a la Recerca (PSR) del Departament d'Enginyeria de Serveis i Sistemes de la Informació.

2.1.4 Desenvolupador

Joan Colom Montoya, estudiant de la FIB i autor del document.

2.1.5 Usuaris

Els usuaris finals del sistema de monitorització seran empreses que requereixen dels serveis proporcionats per la API a nivell de màrqueting i Social Media. La API serà una eina que els ajudarà a gestionar les xarxes socials, saber quin és l'impacte de les campanyes publicitàries i quina és la imatge d'ells que tenen els internautes que utilitzen aquestes xarxes.

Els usuaris que faran servir l'API amb la finalitat de dotar de totes aquestes característiques indicades i altres a nivell analític seran els desenvolupadors de les empreses interessades. Seran el pont entre els serveis que proporciona la nostra API i una aplicació fàcil d'utilitzar per a persones amb coneixements informàtics menys tècnics.

Tot i que estimem que aquests seran la major part dels usuaris finals, l'API està oberta a qualsevol persona que vulgui fer-la servir, per qualsevol mena d'ús lícit que se'n vulgui fer.

Serà necessari que l'usuari disposi d'un compte d'empresa a Instagram, lligat a un compte d'empresa de Facebook, doncs Instagram és propietat de Facebook i és aquesta última qui gestiona tot el tema d'APIs. És possible fer-ho sense ser una empresa, doncs per realitzar aquest TFG s'ha fet.

2.2 Estat de l'art

2.2.1 Contextualització

Les tendències dels consumidors (població) estan canviant. Cada cop és més freqüent que les persones consumeixin menys contingut pels mitjans tradicionals (TV, radio, revistes, diaris, etc.) i en consumeixin més de digitals, com plataformes de contingut multimèdia o xarxes socials.

Els mitjans tradicionals sempre han intentat generar contingut (de menor o major qualitat) que satisfaci els interessos i/o gustos de la seva audiència.

El nombre d'usuaris de les xarxes socials amb múltiples propòsits creix any rere any i, enguany, és molt gran [1].

Amb l'aparició de les noves tecnologies, la tendència ha canviat.

- Hi ha **plataformes de distribució de contingut multimèdia** d'interès pels usuaris, que el posen a disposició de l'usuari final en el moment que aquest desitja (Spotify, Apple music, Netflix, HBO, Amazon prime vídeo, ...), entre altres facilitats.
- **Xarxes socials**, que acostumen a generar contingut de "poca" qualitat (posem entre cometes perquè és d'interès pel seu entorn com a tafaneria, no és de qualitat però al seu entorn li interessa) per l'entorn de cada usuari de la xarxa, però que en general, no és de qualitat ni d'interès pel món en general.

El fet és que, la gent opina moltes coses per les xarxes socials, què és l'àmbit en que ens centrarem. Això propicia que la gent opini sobre tot, ja siguin marques, polítics, estrelles mediàtiques, vídeos o imatges virals, etc.

L'eina de la qual estem parlant en aquest document, es centra amb el què diuen (o altres paràmetres de les publicacions, com imatges o vídeo) dels usuaris en múltiples xarxes socials sobre temes d'interès per quelcom en particular (marques, polítics, estrelles mediàtiques, ...). L'usuari final que fa servir l'eina pot triar alguns paràmetres per filtrar i així, obtenir informació més àmplia on mirar, és a dir, imatges, textos, etiquetes, localitzacions, comentaris, ...

Cal esmentar que tot i que l'eina es centra amb la opinió que tenen els usuaris en múltiples xarxes socials, l'objectiu de la mateixa, no és pas centrar-se en el *sentiment analysis* [2]. És més aviat una finalitat que un tercer pot trobar-hi.

Aquestes últimes esmentades, tenen una oportunitat molt gran a nivell de màrqueting. S'ha passat del tradicional boca a boca a tenir una eina potencial a les xarxes socials. Les empreses poden aprofitar les xarxes socials per reactivar la seva marca: escoltar, segmentar, parlar, mobilitzar, ajudar i involucrar als possibles clients per convertir-los en usuaris fidels. Tot generant publicitat i campanyes personalitzades a cada client [3]. Addicionalment, amb l'aparició dels telèfons intel·ligents i la integració que tenen amb les xarxes socials, ha aparegut un nou tipus de comportament cap a les xarxes socials. Aquest és l'addicció que senten els usuaris vers aquesta combinació. [4].

Un altre ús de les xarxes socials és detectar esdeveniments o catàstrofes sobrenaturals que poden afectar a un conjunt de la població concentrada en una àrea determinada. Hi ha eines que es dediquen a utilitzar la IA (decisiones preses de manera similar a com ho faria un ésser viu), que tot combinant-la amb el monitor de xarxes socials, poden desenvolupar una eina per detectar si han ocorregut tragèdies i si els usuaris de la xarxa social estan bé. Això es pot fer de manera directa o indirecta, ja fan aquesta funció. Quan diem que ho fan directament, és perquè tenen una eina específica que et pregunta si escau [5]. Altrament, si ho fan indirectament, és perquè els usuaris utilitzen aquesta eina per comunicar que estan bé sense que la plataforma tingui cap eina específica [6]. Una altra vesant respecte les anteriors citades, és el d'eines que s'utilitzen per monitoritzar les xarxes socials i fer-ne un anàlisi psicològic dels seus usuaris en funció de les seves publicacions. Concretament, es va fer un anàlisi de la satisfacció dels usuaris amb la seva imatge i les relacions amb les seves respectives parelles a partir de les imatges publicades a Instagram segons el número d'imatges de tipus selfie que es feien [8]. Això es va poder fer recol·lectant la informació sobre els selfies que pujaven a Instagram, amb la etiqueta *#selfie* i passant un qüestionari a posteriori. Aquest és un clar exemple de l'anomenat *sentiment analysis*, el qual no entra en l'abast del projecte, però si que es pot fer servir l'eina desenvolupada i esmentada durant tot el treball per fer diferents tipus de *sentiment analysis*, tot automatitzant gran part de les tasques.

2.2.2 Estudi de mercat

Hi ha moltes eines amb funcions similars. A continuació, n'anomenarem les més rellevants i descriurem què les diferencia.

2.2.2.1 Hotsuite [9]

Està considerada una de les millors plataformes de monitorització de xarxes socials (35 en total). Té unes funcionalitats de gestió de xarxes molt àmplies.

També ens dona la opció de generar tasques i assignar-les, enviar missatges privats i canviar l'estat de les tasques. Les darreres funcions en qüestió, són molt útils per coordinar la gestió del perfil digital de les empreses, persones mediàtiques o famoses, etc.

A més, permet descarregar els balanços periòdics de les diferents xarxes socials. Per finalitzar, també ens dona la possibilitat de formar-nos en aquesta eina / plataforma i certificar-nos en diferents àrees (en la plataforma, en màrqueting social, estratègia dels medis socials avançats, ventes i el d'una clínica en concret).

2.2.2.2 Brandwatch [10]

Aquesta és una eina de pagament molt utilitzada a nivell empresarial. Bàsicament, permet veure quina influència tenen les teves accions a les xarxes socials, quins són els usuaris de les xarxes que estan més orientats cap al teu sector, a quina hora exerciràs més influència sobre ells amb les teves accions i visualitzar estadístiques que t'ajudaran a la presa de decisions a nivell empresarial.

2.2.2.3 Media monitoring API de mention [11]

Aquesta API és molt completa i avarca més enllà del que fa la nostra API. Aquesta també fa una monitorització dels mitjans.

Adicionalment, permet ser integrada en aplicacions de tercers.

No obstant, té un sistema de subscripcions que amb diverses modalitats d'usuari:

- Versió de prova gratuïta per un temps limitat.
- Compte individual per 29 €/mes.

- Compte de debutant per 99 €/mes.
- Compte empresa.

En cadascuna d'aquestes versions, es va incrementant el preu i les prestacions disponibles. Es van incrementant el nombre d'alertes a rebre, les mencions que es poden fer i el nombre d'usuaris per compte, de manera que diversos usuaris poden gestionar el compte.

2.2.3 Conclusions de l'estudi de mercat

Degut a que les xarxes socials estan força arrelades a la nostra cultura i, que la monitorització de les xarxes socials a nivell empresarial van començar a agafar empena fa un parell d'anys, estem davant d'un mercat força competitiu.

Hi ha moltes eines de monitorització, gairebé totes tenen un pla de prova i diversos de pagament. En funció d'això, es té accés a més o a menys funcionalitats.

El nostre sistema està enfocat a veure quines són les respostes donades a els nostres publicacions, les mencions que se'ns fan i podem filtrar per paraula clau. D'aquí es pot trobar la utilitat de saber quins són els sentiments de les persones cap a certes coses (*keywords* o paraules clau), ja sigui a nivell polític, davant d'una catàstrofe natural o qualsevol altre.

Un dels principals motius pels quals no volem utilitzar solucions ja existents és el fet que, tot i que n'hi ha que són gratuïtes, no són *open source*, i a la que volem fer més del que es considera bàsic (com ara, interactuar amb la seva API i fer un nombre elevat de peticions), o no se'ns deixa, o bé se'ns fa passar a un pla de pagament.

En tot cas, els serveis existents són tals que no estan orientats a desenvolupadors que vulguin crear quelcom nou partint de la seva creació com a part de la nostra pròpia solució.

El principal tret que fa que el nostre sistema es diferenciï de la resta que hi ha actualment al mercat o per la web és la combinació d'aquestes tres característiques:

- És *open source*.
- Està desenvolupat com a servei web per facilitar la integració amb altres components.

En altres sistemes, es compleixen aïlladament, però no de forma conjunta.

3 Formulació del problema i abast

El sistema resultant d'aquest projecte esdevindrà resultat de la integració de:

- Monitoritzar els comentaris que es fan a la pròpia compte d'Instagram i les mencions que se'ns fan a les nostres publicacions.
- Poder filtrar les dades monitoritzades per paraula, tal que els resultats monitoritzats hauran de satisfer el fet de contenir la paraula clau en el seu text.
- Tenir un projecte fàcil de mantenir i ampliar per tal de poder obtenir més metadades rellevants del que busquem (localització, enllaços a les fotografies d'instagram, etc.).

3.1 Objectius

3.1.1 Objectius principals

L'objectiu que s'ha de complir en el nostre sistema per tal que es resolgui el problema formulat anteriorment.

- Donar una solució software que permeti obtenir les dades i metadades rellevants dels missatges que compleixin amb el nostre criteri de cerca (per exemple, l'usuari que ens ha fet el comentari o menció, l'hora exacta, número de m'agrada, etc.).

3.1.2 Objectius específics

- Anàlisi dels requeriments del sistema de monitorització.
- Disseny de la nova mantenint l'alta cohesió i baix acoblament per a possibles ampliacions, com ara formar part del sistema del qual parteix la idea d'aquest TFG.
- Desenvolupament dels esquemes conceptuals que permetin emmagatzemar les dades monitoritzades.
- Implementació de l'eina.
- Validació del sistema.

3.2 Abast

Les tasques que queden dintre de l'abast del nostre projecte són:

- Monitoritzar Instagram.

A l'abast també és important definir aquelles coses on no s'arribarà:

- Monitor de facebook degut a l'impediment tecnològic extern a nosaltres.
- Monitorització de contingut privat dels usuaris de les xarxes socials, per exemple, missatges de Facebook Messenger, WhatsApp o missatges privats dins d'Instagram.
- Agregació de tota la informació per a fer anàlisi estadístic (el nostre sistema genera dades que es poden fer servir per tercers que vulguin implementar aquesta funcionalitat mitjançant aquest com a component d'altres sistemes).
- *Sentiment analysis* a partir de les dades monitoritzades.

3.3 Possibles obstacles

En aquest apartat, tractarem els possibles obstacles que ens podem trobar durant el desenvolupament del nostre projecte segons la tipologia dels obstacles.

3.3.1 Desconeixement de la tecnologia

Les tecnologies que faig servir per realitzar el projecte són més o menys noves en el sentit que:

- Algunes no les he fet servir mai.
- Altres les conec a nivell teòric però no a la pràctica.
- D'altres, tinc poc coneixement pràctic o coneixement en tecnologies homòlogues.

3.3.2 Impediments tecnològics causats per la API externa

Ens hem trobat que hi ha diverses xarxes socials que són molt estrictes en quant al nombre de peticions que se'ls pot fer i que imposen polítiques molt restrictives.

Per exemple, la API *Instagram Platform API* (2), que és per a comptes que no són negocis, ens diuen que com a política, que no abusem de la plataforma de la API, ni automatitzem peticions o fem ús d'un mal comportament. Aquestes accions ens revocarien l'accés.

Un altre exemple de política de restricció és la del *Facebook Graph API*, que quan intentes agafar dades d'usuaris qualsevols, et deixa agafar certes dades (com el identificador de facebook, la imatge de l'usuari i el seu nom i cognoms). En canvi, quan es vulguin obtenir altres dades (com el contingut de les publicacions dels usuaris),

revisaran primer el teu codi de manera exhaustiva abans de donar-te permís per registrar una API que faci el que desitges.

- Possible solució: documentar-se en els fòrums d'internet si aquests són problemes recurrents en l'actualitat. En cas afirmatiu, mirar altres xarxes socials amb APIs que no tinguin aquestes restriccions.

3.3.3 Manca de temps

Un altre obstacle podria ser la manca de temps per part del desenvolupador. Sempre es desitjable que no hi sigui aquest inconvenient, però a vegades hi ha fets que escapen a tota planificació (com malaltia, falta de la disponibilitat de l'entorn de treball, etc.).

- Possible solució: seria bo tindre una mica de tolerància a l'error amb la planificació temporal, deixar alguns objectius com a obligatoris i alguns altres com a opcionals. Si es dona el cas que no hi ha hagut cap imprevist que hagi afectat el temps i es poden assolir més objectius (dels opcionals).

3.3.4 Catàstrofes tecnològiques

Sempre es pot produir una pèrdua, ser víctimes d'un robatori o que s'espantin els equips del nostre entorn de treball.

- Possible solució: tindre còpies de seguretat en dispositius externs, al núvol i fer el possible per poder seguir treballant ràpidament en cas de que el nostre equip esdevingui en no funcional.

3.3.5 Problemes de salut

Degut al meu historial mèdic i al meu passat en termes referits a aquest tema, pot ocórrer que hagi d'interrompre el TFG per aquesta qüestió.

Sense anar més lluny, el darrer any vaig haver de fer una pausa per motius de salut, vaig estar gairebé tot el 2018 amb recaigudes recurrents.

4 Metodologia i rigor

Durant la realització del TFG ens trobem amb el fet de que s'han d'assolir molts objectius, coneixements i metes en un període de temps curt i uns recursos humans limitats.

Aquestes circumstàncies ens porten a adoptar una metodologia de treball àgil. Així, el treball que es realitzi serà comprovat pel director / codirector / professor de GEP (la part pertinent) tot sovint. Tanmateix, es mantindrà una comunicació fluida i les reunions seran freqüents.

Degut a que el TFG és un treball individual i els fets esmentats anteriorment, farem servir la metodologia de treball àgil KANBAN.

En conclusió, no donarem peu a un gran desviament dels nostres deures degut a qualsevol error comès i l'impacte que ocasiona el temps sobre el mateix.

Primer es definiran els objectius i requisits, per després extraure'n les històries d'usuari. Seguidament, es dividiran les històries d'usuari en subhistòries o tasques, si escau, per poder fer una estimació del temps que ocuparà cadascuna de les històries d'usuari o tasques (la planificació temporal es troba al punt 5 del document).

4.1 Introducció a KANBAN

KANBAN és una metodologia àgil que va sorgir a la fàbrica japonesa de Toyota com a resposta a ser més productius quan l'estoc dels components era escàs i impedia seguir avançant en el treball [12].

Com és ben sabut, tenir estoc innecessari al magatzem de qualsevol fàbrica és un sobrecost. Aquest obstacle es pot vèncer amb el sistema *Just In Time* (JIT) que es proposa a la metodologia en qüestió [13].

A més, és una metodologia ja provada en el camp del desenvolupament de programari i s'ha comprovat que és una metodologia que s'adapta als requeriments d'aquest camp [14].

Es realitza una producció sense malbarataments i es pot adaptar a les necessitats específiques. Si hi ha quelcom que no funciona correctament al KANBAN, es canvia segons els requeriments de la persona i/o projecte [15].

4.2 Explicació de la metodologia KANBAN

En aquest apartat explicarem com funciona la metodologia i com es porta a la pràctica. L'element principal de KANBAN és una taula o taulell que té 5 columnes amb capçaleres:

- TODO: descripció de les tasques a fer, ordenades segons la importància que tenen pel projecte.
- Anàlisi: s'analitzen els requisits de la tasca i es veu si és viable o si és millor separar-la en tasques més petites.
- Disseny: disseny de la tasca / funcionalitat(s).
- Implementació: construcció a nivell software.
- Testing: integració i proves.
- Completat: les tasques que ja s'han fet, passen aquí.

Les tasques que anomenem, són *post its* o notes adhesives que es poden anar movent per les files i les columnes.

Quan realitzem una tasca en una columna (o fase), passarà a la següent columna a sota de tot, ja que està ordenat per prioritats. Tot i això, cal recordar que podem adaptar la metodologia a les nostres necessitats, doncs els projectes són dinàmics i poden vindre nous requisits preferents en qualsevol moment [16].

Per acabar, també cal comentar que aquesta metodologia imposa una restricció, que pot ser definida segons les característiques específiques de cada projecte. Aquesta consisteix en que en cadascuna de les columnes intermèdies de la taula, es fixa el màxim de tasques que hi poden haver. Això es fa per evitar tenir colls d'ampolla i que s'acumuli la feina. Així aconseguim que el mètode ens aporti productivitat.

4.3 Iteracions setmanals o cada dues setmanes

En principi, les iteracions seran d'1 setmana degut als motius argumentats anteriorment. Segons la càrrega de treball de les històries d'usuari a implementar i de la documentació pertanyent, es poden allargar.

4.4 Reunions (semi)presencials setmanals o cada dues setmanes

S'intentarà que les reunions siguin setmanals. D'aquesta manera, hi ha el *cara-a-cara* i les reunions són més productives, s'entén tot millor i és més fàcil proporcionar el *feedback* sobre el treball fet.

En el cas de festivitats o que un o diversos dels membres citats als apartats [2.1.2](#), [2.1.3](#) i [2.1.4](#) no puguin assistir, es podria posposar la reunió i seguir amb la planificació establerta (sempre prèvia comunicació mitjançant les eines disponibles).

Tot el que s'ha esmentat en aquest punt anteriorment, és el que s'ha realitzat durant bona part del projecte, però degut a canvis d'habitatge, s'ha canviat a un seguiment telemàtic a distància, amb eines de videoconferència i compartició de pantalla.

4.5 Eines de seguiment

Per realitzar el seguiment del projecte software es farà servir un repositori GIT.

També tenim la taula del KANBAN, que ens ajuda a visualitzar de manera ràpida l'estat de les tasques a realitzar, així com veure les que s'han completat.

Per acabar, disposarem del diagrama de Gantt, amb el qual podrem veure per quina fase del projecte estem i si cal donar-se presa degut a dates límit.

En cas que no funcionés o que hi hagués hagut alguna demora d'acord amb la planificació temporal, s'haurien de fer estimacions noves degut al contratemps.

4.6 Mètode de validació

Durant les reunions i quan s'accepta la petició de pujada de codi a la branca principal del repositori GIT, es comprova que no hi ha problemes en el codi ni a les funcionalitats implementades.

Per comprovar que la feina feta és correcta, s'executaran tests mitjançant les crides desades al *Postman* i es validarà el correcte funcionament dels monitors, tant per les característiques funcionals (execució sense errors, resultats de la monitorització correctes i esperats, etc.) com no funcionals (rendiment, consum de recursos, etc.).

Com que aquests tests es fan de forma periòdica a mesura que es van fent les implementacions, sense deixar molts canvis en la implementació entre mig, és fàcil identificar quins són els possibles problemes i on estan localitzats.

5 Planificació temporal

5.1 Estimació temporal del projecte

Aquest projecte va començar el setembre del 2017 amb la part de GEP, però degut a problemes personals, no es va reprendre fins al febrer / març del 2019.

S'ha planificat una dedicació de 30 hores setmanals, comptem amb 540 hores per a la realització del projecte.

Considerem que una setmana té 7 dies i es treballa en cadascun d'ells. Al final de la setmana, s'hauran fet una mitjana de 30 hores (4 hores, 17 minuts i 20 segons).

No obstant, hi haurà algunes setmanes que requeriran més hores i d'altres que en requeriran menys, segons les tasques que s'hagin de dur a terme en aquell moment.

5.2 Recursos

Els recursos que s'utilitzaran per a la realització del projecte es poden classificar en dos tipus: personals i materials.

5.2.1 Recursos personals

Com a recursos personals, es disposa d'una persona per a realitzar el projecte, l'espai de treball i les 30 hores setmanals que es dedicaran al projecte durant el període de temps que ocupa el projecte.

5.2.2 Recursos materials

Podem classificar els recursos materials com a hardware i software.

Pel que respecte als recursos materials de tipus hardware, tenim:

- Ordinador portàtil Macbook Pro midle 2018: per a realitzar tot el projecte, des de les tasques de redacció i gestió, fins a les tasques d'implementació i validació del producte.
- Pantalla, teclat i ratolí: per a poder treballar més còmodament quan realitzo el treball en el meu entorn de treball habitual.
- Discs durs externs: per a realitzar còpies de seguretat a tot el treball.
- Espai d'emmagatzemament *hosting* per a tenir-hi l'API d'Instagram, encara a concretar.

Respecte als recursos software / programari utilitzats:

- Suite d'ofimàtica Microsoft Office 2016: per a la redacció de document, fulls de càlcul per realitzar tasques relacionades amb la gestió temporal i econòmica.
- Eclipse JEE 4.6: eina de desenvolupament de programari que s'utilitzarà per a dur a terme la implementació de tot el programari realitzat en el projecte. Aquesta interfície / entorn de desenvolupament incorpora suport per a sistemes de control de versions, com GIT, el qual utilitzarem.
- Postman 5.2.1: eina per a realitzar les invocacions als mètodes suportats per l'API d'Instagram, i comprovar el resultat d'aquestes.

5.3 Pla d'acció i alternatives

En cas de que s'observi que no s'assoleixen les dates límit de les diferents tasques, es provarà de treure partit als avantatges que ens proporciona la metodologia KANBAN.

Cada tasca que sigui susceptible de causar-nos problemes en el compliment de les dates, o que sigui de risc mitjà / alt, es dividirà en tasques més petites i de menor granularitat. S'ordenaran per ordre d'importància pel màxim assoliment de les funcionalitats i objectius del projecte.

Aquesta acció ja simplificarà bastant la feina a realitzar, i en cas que alguna de les tasques noves quedi pendent, serà de menor importància i no posarà el projecte en un risc de perillositat elevada.

5.4 Calendari

5.4.1 Estimació d'hores i dates

Tasques	Inici	Duració (dies hores)	Final	Risc
1 Definir objectius i metodologia del projecte	19-09-17	7 18	26-09-19	Mitjà
2 Planificació temporal	26-09-17	6 15	02-10-17	Baix
3 Gestió econòmica i sostenibilitat	02-10-17	7 12	09-10-17	Baix
4 Presentació preliminar	09-10-17	7 10	16-10-17	Mitjà
5 Definició i verificació de les condicions	16-10-17	7 10	23-10-17	Baix
6 Presentació oral i document final	13-10-17	7 4	23-10-19	Mitjà
7 Instal·lació entorn	21-02-19	7 12	28-02-19	Baix
8 Aprendre tecnologies a usar (API, JSON, ...)	28-02-19	14 15	14-03-19	Mitjà
9 Anàlisi de requeriments	14-03-19	7 18	21-03-19	Alt
10 Estudi de l'arquitectura actual	21-03-19	21 60	11-04-19	Mitjà
11 Disseny i implementació dels canvis arquitectònics	12-04-19	21 85	03-05-19	Alt
12 Disseny i implementació nou monitor Instagram	03-05-19	32 68	04-06-19	Alt
13 Testing del monitor d'Instagram	06-06-19	1 4	07-06-19	Mitjà
14 Disseny i implementació nou monitor Instagram (delay)	21-08-19	28 80	18-09-19	Alt
15 Testing del monitor d'Instagram (delay)	19-09-19	14 40	03-10-19	Mitjà
16 Redacció de la memòria del TFG	29-04-19	46 26	14-06-19	Mitjà
17 Validació final	27-09-19	14 60	10-10-19	Mitjà
18 Preparació de la presentació final	04-10-19	7 30	18-10-19	Mitjà

Taula 1 Estimació d'hores i dates

Com podem veure en aquesta estimació d'hores i dates, hi ha tasques que es realitzen alhora que altres. No vol dir pas que es facin realment alhora, sinó que KANBAN ens permet anar fent en cada moment parts de tasques diferents i amb el marge d'hores que gestionem, és factible.

Gràcies a la flexibilitat de KANBAN, podem adaptar-lo a les necessitats de la persona que realitzarà el projecte. En el meu cas concret, no m'agrada estar fent un tipus de feina durant dies fins que l'acabi mentre n'hi ha d'altres pendents (i en això aporta molt de sentit la metodologia aplicada). Prefereixo fer unes hores una tasca i unes quantes una altra. Així completo la fase d'una tasca de KANBAN i moc la nota adhesiva de la taula de KANBAN cap a la següent fase. Seguidament, puc realitzar feina d'una altra tasca segons les necessitats del projecte.

Degut als motius explicats, quan dic que una tasca dura 7 dies, però 20 hores, (per exemple, a la tasca número 10) vol dir que mentre faig aquella tasca també en faig una altra durant aquella setmana.

Les tasques que tenen “(delay)” al final del seu nom són tasques que s'han hagut de partir degut a que no s'ha pogut establir la continuïtat durant un període de temps, causat per imprevistos. Tot i que els imprevistos no haurien d'aparèixer a dins de la planificació temporal, quan s'ha de refer la planificació temporal (degut a que es duia massa retràs per poder presentar el TFG al torn de lectura anterior) i un es troba amb un imprevist amb perspectives a futur, es té en compte.

5.4.2 Dates importants al calendari

Les dates importants van lligades a la normativa del TFG. Aquestes influencien el calendari intern que hem definit per a realitzar les diferents tasques a realitzar de les que consta el nostre projecte.

A continuació indiquem quines són aquestes:

- 27/09/19 Elegir torn de lectura (data límit).
- 27/09/19 Reunió de seguiment.
- 14-18/10/19 Presentació de la memòria amb un mínim d'una setmana d'antelació respecte la data acordada per la defensa.
- 21-25/10/19 Lectura del TFG.

5.4.3 Descripció de les tasques

1 Definir objectius i metodologia del projecte: redactat dels objectius que ha d'assolir el projecte i de quina metodologia es farà servir per realitzar tot el projecte i poder tenir

èxit amb el projecte. Això és molt important, ja que s'ha de ser realista segons els recursos disponibles i triar una metodologia que encaixi amb els recursos i el temps disponibles.

2 Planificació temporal: es defineixen les tasques que s'han de realitzar i s'estableix el calendari per a realitzar-les.

3 Gestió econòmica i sostenibilitat: s'identifiquen els recursos necessaris per a realitzar el projecte i s'estima el cost que tindrà. Tanmateix, es diu quin és el possible impacte ambiental, econòmic i social que tindrà el projecte.

4 Presentació preliminar: es prepara una presentació on es donen a conèixer totes les dades rellevants recopilades fins al moment.

5 Definició i verificació de les condicions: juntament amb el director i codirector del projecte, és l'hora de repassar quines són les competències que toca el projecte. Ja s'havia fet en un primer moment, però ara es té cert bagatge per afirmar amb certesa quins seran, si cal fer alguna modificació, etc.

6 Presentació oral i document final: es recopila tota la informació presentada fins al moment i es fa una curta exposició de 5 minuts.

7 Instal·lació de l'entorn: es preparen els dispositius electrònics (ordinador, servidor) per a realitzar el projecte, amb tot el programari necessari.

8 Aprendre tecnologies a usar (API, JSON, ...): investigació de com funcionen les tecnologies necessàries per realitzar el projecte.

9 Anàlisi de requeriments: s'estudien totes les funcionalitats que ha de complir el projecte per a que, un cop estigui realitzat, es pugui considerar que és un projecte que satisfà els objectius definits.

10 Estudi de l'arquitectura actual: degut a que aquest és un projecte que parteix d'un altre existent, el desenvolupador ha d'estar familiaritzat amb l'arquitectura actual, que marcarà com han de ser les ampliacions a nivell arquitectònic.

11 Disseny i implementació dels canvis arquitectònics: degut a que el projecte actual funciona per a monitoritzar una única xarxa social, es volen aplicar els canvis necessaris per a poder monitoritzar altres xarxes socials.

12 Disseny i implementació del nou monitor d'Instagram: disseny a nivell arquitectònic que el monitor d'Instagram, i posterior implementació.

13 Testing del monitor d'Instagram: proves per verificar que el monitor d'Instagram satisfà el requisits específics a aquest.

14 Disseny i implementació del nou monitor d'Instagram (delay): idèntic a la tasca 12 però s'ha hagut de partir per problemes amb la planificació (veure final de l'apartat 5.4.1 [Estimació d'hores i dates](#)).

15 Testing del monitor d'Instagram (delay): idèntic a la tasca 13 però s'ha hagut de partir per problemes amb la planificació (veure final de l'apartat 5.4.1 [Estimació d'hores i dates](#)).

16 Redacció de la memòria del TFG: recollida de totes les dades que es necessiten per documentar la solució a la qual arribem per donar resposta al problema plantejat pel TFG.

17 Validació final: es verifica que la integració dels diversos monitors dóna cabuda a un projecte final que satisfà els requisits definits i aporta una solució correcta al problema plantejat a l'abast del TFG.

18 Preparació de la presentació final: es prepara la defensa del nostre TFG, que es durà a terme davant del tribunal corresponent.

5.4.4 Dependències entre tasques

Les dependències d'una tasca B respecte una A, ens indica que no es pot iniciar la tasca B si abans no s'ha completat la A. Les dependències que tenim són:

- 2 Planificació temporal → (depèn de) 1 Definir objectius i metodologia del projecte: es necessiten els objectius per poder fer la tasca 2.
- 3 Gestió econòmica i sostenibilitat → 2 Planificació temporal: necessitem saber les tasques i quant duraran per poder saber les hores que requerirà cada tasca i poder fer l'estimació econòmica segons les hores i el rol de la persona que la realitza.
- 9 Anàlisi de requeriments → 5 Definició i verificació de les condicions: per fer l'anàlisi de requeriments es necessita haver verificat quines seran les condicions que ha de complir el projecte, perquè poden haver canviat del que es tenia pensat en un primer moment.

- 11 Disseny i implementació dels canvis arquitectònics → 7 Instal·lació de l'entorn
8 Aprenentatge tecnologies: abans de construir, necessitem saber les eines i com utilitzar-les.
- 15 Testing del monitor d'Instagram (delay) → 14 Disseny i implementació del nou monitor d'Instagram (delay), doncs per poder començar a fer testing, necessitem una mínima feina feta a la implementació del nou monitor. No necessitem que estigui tota la tasca completada com diem a l'inici d'aquest apartat, però si que hi ha una clara dependència perquè no es pot dur a terme la tasca 15 sense haver fet res de la 14, i la tasca 15 es completarà necessàriament després de la tasca 14. Gràcies a la metodologia KANBAN (veure apartats [Introducció a KANBAN](#) i [Explicació de la metodologia KANBAN](#)) podem suplir dues tasques seqüencials per les mateixes en paral·lel.
- 17 Validació final → 15 Testing del monitor d'Instagram (delay): sense un resultat satisfactori de la tasca 15, la tasca 17 té un resultat incorrecte.
- 18 Preparació de la presentació final → 16 Redacció de la memòria del TFG 17 Validació final: per preparar la defensa, s'han d'haver realitzat totes les altres tasques del projecte.

5.4.5 Diagrama de Gantt

A continuació es mostra el diagrama de Gantt del nostre projecte.

A la primera columna, es mostra el nom de la tasca. Les hem dividit en tres blocs, on s'hi engloben la majoria, tal i com es pot apreciar.

A la barra superior del diagrama tenim les dates en que es realitzarà cadascuna de les tasques (*a grosso modo*). Per concretar les dates exactes, podem consultar la taula de l'apartat 5.4.1 [Estimació d'hores i dates](#).

Cal mencionar uns canvis que pot no entendre's al diagrama.

Inicialment, es va fer GEP al 2017 però degut a problemes de l'estudiant, un cop acabat GEP, no es va reprendre fins l'any 2019.

Després, es va ser planificat per ser presentat a la lectura del torn de juliol d'aquest mateix any, però vaig tenir molts problemes amb el dissenys i implementació del monitor d'Instagram. Degut a això he posat les mateixes tasques però com a (*delay*).

Aquesta nomenclatura també ha estat motivada perquè no es podia allargar la tasca anterior degut a un lapse de temps important pel mig, causat per més imprevistos.

Degut a tot això, hem mostrat el diagrama de Gantt en dues parts. D'aquesta manera és tot més entenedor a nivell de visualització. Altrament, es veuria un lapse molt llarg pel mig. L'eina utilitzada per fer el diagrama de Gantt no ens permet posar un “*parèntesis temporal*”.

Enllaç i QR del diagrama de Gantt de la part de GEP per veure'l millor:

[Enllaç diagrama de Gantt GEP 2017](#)



Fig 1 Codi Qr amb l'enllaç al diagrama de Gantt de GEP

A la següent pàgina es mostra el diagrama de Gantt per la part de GEP feta l'any 2017.

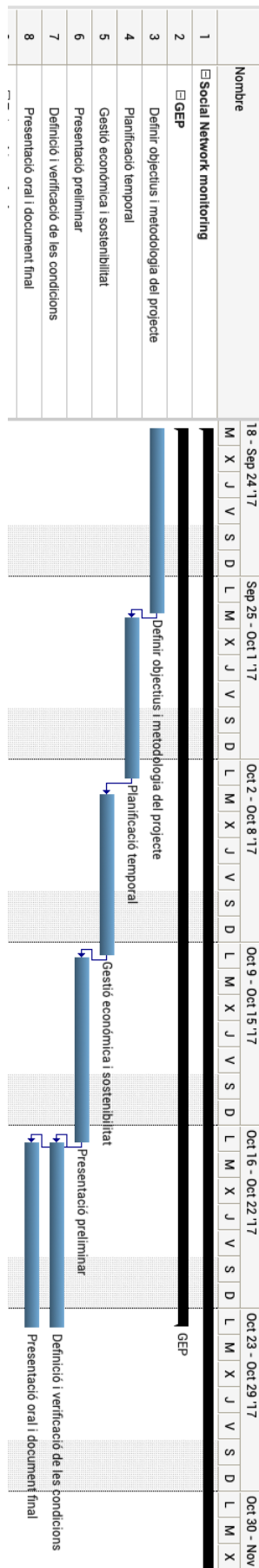


Fig 2 Diagrama de Gantt de GEP

Enllaç i QR del diagrama de Gantt de la part de la resta del TFG per veure'l millor:

[Enllaç diagrama de Gantt TFG part 2019](#)



Fig 3 Codi Qr amb l'enllaç al diagrama de Gantt de la resta del TFG

A la següent pàgina es mostra el diagrama de Gantt per la part de la resta del TFG feta l'any 2019.

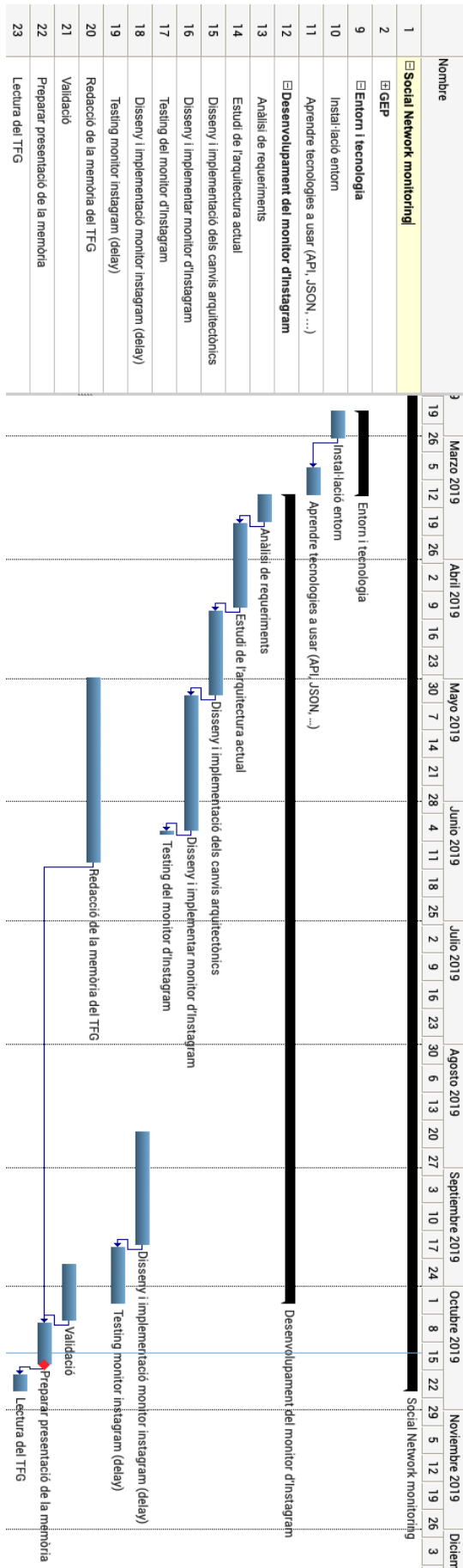


Fig 4 Diagrama de Gantt de la resta del TFG

6 Gestió econòmica

6.1 Consideracions a tenir en compte

A continuació, presentem l'apartat econòmic del projecte. En tot projecte on s'ha arribat a aquest punt, és important que abans de procedir i avançar, es faci una estimació del pressupost i dels beneficis. Els beneficis interessin quan hi ha *shareholders*. En aquest cas concret, no n'hi ha, ni inversió econòmica aliena a l'equip de desenvolupament.

Tampoc es tracta d'un projecte amb fins econòmics, no es busca retorn de la inversió (ROI), les eines que s'utilitzen són (en la seva majoria) *open source*, i l'objectiu és produir una eina *open source*. Per tant, no es farà negoci.

6.2 Identificació i estimació dels costos

Per començar, identificarem els costos dels recursos humans i en justificarem el seu cost unitari.

A la següent taula, podem observar els costos dels diferents rols de manera unitària (€/hora), el nombre total d'hores que hi dedica i el cost total de cada rol.

Rol	Sou/hora	Total hores	Cost total
Cap de projecte	27,39 €	125	3.423,75 €
Analista	24,90 €	18	448,20 €
Dissenyador	22,91 €	153	3.509,81 €
Programador	16,19 €	166	2.700,49 €
Tester	19,92 €	104	2.071,68 €
Total	-	567	12.153,93 €

Taula 2 Cost total per rol

Els diferents salaris estan basats en el salari mig dels diferents rols l'any 2016 amb una experiència d'entre 3 i 7 anys [17].

Els salaris dels costos totals que apareixen en aquesta pàgina són anuals, per obtenir el salari per hora, he considerat que es treballen 251 dies a l'any i que les jornades són de 8 hores. Aquests salaris son bruts, sense impostos.

Aquest TFG és realitzat per un estudiant de Grau en Enginyeria informàtica, no pas per persones amb els rols i antiguitat indicats a la taula superior. Tot i que la meva categoria real seria de programador junior o becari, hi ha una gran diferència respecte al que succeiria si estès desenvolupant aquest projecte de forma completament autònoma. I és, doncs, que tinc el suport de professionals amb àmplia experiència en el sector (professorat i altre personal docent de la FIB).

El suport que rebo és en forma de guia a seguir, resolució de dubtes que pugui tenir i feedback setmanal sobre tot el treball que vaig generant.

Sense cap mena de dubte, aquest és un gran tret diferenciador respecte a la majoria de professionals que treballen de manera autònoma.

6.2.1 Costos directes per activitat

Tot seguit, visualitzarem la taula amb els costos per activitat (la anterior desglossada).

	Activitats i tasques	h	Rol	€/h	Cost total (€)
1	Definir objectius i metodologia projecte	18	Cap de projecte	27,39	493,02
2	Planificació temporal	15	Cap de projecte	27,39	410,85
3	Gestió econòmica i sostenibilitat	12	Cap de projecte	27,39	328,68
4	Presentació preliminar	10	Cap de projecte	27,39	273,90
5	Definició i verificació de les condicions	10	Cap de projecte	27,39	273,90
6	Presentació oral i document final	4	Cap de projecte	27,39	109,56
7	Instal·lació entorn	12	Programador	16,19	194,28
8	Aprendre tecnologies (API, JSON, ...)	15	Programador	16,19	242,85
9	Anàlisi de requeriments	18	Analista	24,90	448,20
10	Estudi de l'arquitectura actual	60	Dissenyador	22,91	1374,60
11	Disseny i implementació dels canvis arquitectònics	85	Dissenyador → 40% Programador → 60%	18,88	1604,63
12	Disseny i implementació d'un nou monitor per Instagram	68	Dissenyador → 40% Programador → 60%	18,88	1283,70
13	Testing del monitor d'Instagram	4	Tester	19,92	79,68
14	Disseny i implementació d'un nou monitor per Instagram (delay)		Dissenyador → 40% Programador → 60%	18,88	1510,40
15	Testing del monitor d'Instagram		Tester	19,92	796,80
16	Redacció de la memòria del TFG	30	Cap de projecte	27,39	821,70
17	Validació final	60	Tester	19,92	1195,20
18	Preparació de la presentació final	30	Cap de projecte	27,39	821,70
	Total	567			12154,09

Taula 3 Costos directes per activitat

Nota aclaridora: inicialment, les tasques 12, 13, 14 i 15 eren només la 12 i la 13. Degut a que no es van assolir els objectius amb el temps acordat i als imprevistos, es van afegir les tasques 14 i 15, així com la modificació de la ubicació en l'espai temporal de les tasques 12 i 13.

A part d'això, considero que no s'ha fet una desviació tan gran respecte la planificació original, doncs han sorgit imprevistos que han dificultat el camí i difícils de superar. Cal tenir en compte que tinc poca experiència, que aquest projecte té una magnitud considerable

6.2.2 Costos indirectes

Durant el transcurs del projecte, aquest serà realitzat des de casa, en gairebé tota la seva totalitat. Això comporta que s'hagi de tenir en compte el preu de la connexió a internet de fibra òptica i el consum elèctric.

Pel que fa el preu del lloguer on realitzo el projecte, no es tindrà en compte perquè és un domicili familiar i no pas un local destinat única i exclusivament a la realització del meu projecte. Es pagaria el mateix tan si fes el projecte com si no.

- Amortització del hardware: s'han d'amortitzar els mesos del portàtil en el que realitzo el projecte, doncs és relativament nou i dista de tenir 4 anys. També s'ha d'amortitzar el cost de la pantalla que utilitzo per connectar-la al portàtil, que és bastant nova, s'haurà d'amortitzar. Els discs durs que utilitzo per fer còpies de seguretat no s'hauran d'amortitzar, doncs es realitzen independentment del projecte.
- Amortització del software: tot el projecte es realitza amb eines *open source*, exceptuant la part d'ofimàtica que la realitzo amb Microsoft Office 2016. La llicència de l'Office la tinc de manera gratuïta fins a finals de gener, doncs me la van proporcionar a la universitat on vaig estar d'Erasmus i té una validesa d'un any a partir de la meua marxa de la universitat destí, és a dir, finals de gener. Per tant, no hauria de tenir cap cost per l'amortització del software.
- Impressions en paper: pel dia de la defensa final del TFG realitzaré 4 impressions de la memòria del TFG. Estimo que, en total, seran un màxim de 400 pàgines a imprimir.

El preu mig del kWh a la factura de la llum és de **0,12411 €/kWh**.

Càlcul de consum elèctric de l'ordinador portàtil:

$$60 \text{ W} \times 567 \text{ h} \times \frac{1 \text{ kW}}{1000 \text{ W}} = 34,02 \text{ kWh} \text{ consumits durant la realització del TFG.}$$

Càlcul del consum elèctric del monitor Dell UltraSharp U2414H:

$$16 \text{ W} \times 567 \text{ h} \times \frac{1 \text{ kW}}{1000 \text{ W}} = 9,07 \text{ kWh} \text{ consumits pel TFG.}$$

Consum elèctric total: 43,09 kWh.

A continuació, veurem la taula que reuneix tots els costos indirectes del projecte:

Producte	Unitats	Preu unitari	Cost estimat	Preu producte	% amortitz.
Electricitat	43,09 kWh	0,12411 €	5,35 €		
Internet	8 mesos	32,00 €	256,00 €		
Amortització pantalla	8 mesos	3,81 €	30,50 €	228,74 €	20 %
Amortització portàtil	8 mesos	46,70 €	373,58 €	2241,47 €	25 %
Discs durs	8 mesos	0,00 €	0,00 €		
Impressions	400 pàgs	0,03 €	12,00 €		
Total			677,42 €		

Taula 4 Costos indirectes

La columna de Cost total i % amortització fan referència als productes amortitzables. En el cas de la pantalla, és un dispositiu electrònic que es pot amortitzar amb un percentatge màxim anual del 20% i un temps màxim de 10 anys. Nosaltres decidim fer-la amb el percentatge màxim anual, és a dir 5 anys.

Pel que fa l'ordinador portàtil, són aparells computacionals, que es permeten amortitzar amb un percentatge màxim anual del 25% i un temps màxim de 8 anys. Decidim fer-la amb el percentatge màxim anual, és a dir, 25% i 4 anys.

El preu unitari dels productes amortitzables és el resultat d'aplicar el percentatge màxim d'amortització anual al preu del producte amortitzat, i dividir-lo per 12 (mesos / any).

6.2.3 Contingència

En tot projecte es recomana deixar 10-15% del pressupost per si hi ha desviacions o imprevistos durant la realització del projecte.

En el nostre cas, decidim escollir un 15% per estar més segurs sobre la viabilitat econòmica del projecte realitzat.

Producte	Cost inicial	% contingència	Cost contingència
Costos directes	12.154,09 €	15%	1.823,09 €
Costos indirectes	677,42 €	15%	101,61 €
Total			1.924,70 €

Taula 5 Costos de la contingència

6.2.4 Imprevistos

Identifiquem dos possibles imprevistos:

- Endarreriment respecte la planificació: en aquest cas, augmentaríem 10 hores setmanals a la dedicació del projecte, arribant a fer l'equivalent a una jornada completa (40h). El màxim d'hores que estimem haver d'ampliar, si es dona la situació, és de 50 hores (~15%).

És important notar que el preu per hora agafat és $(màxim + mínim) / 2$.

- Avaria de l'ordinador portàtil: si succeeix el fet, es comprarà un equip que permeti la restauració de la còpia de seguretat i la ràpida instal·lació de l'entorn mitjançant aquestes còpies. Destinaríem 2030,92 €, que augmentaria l'amortització en 42,31 € mensuals el pressupost.

Els percentatges de probabilitat els hem agafat del *Chaos Manifesto* [18].

A continuació, mostrem la taula on es concreten aquests costos.

Imprevistos	Preu unitari	Quantitat	Cost total	% Probabilitat	Total
Més hores	21,79 €/hora	50 hores	1.089,50 €	50,4	549,11 €
Nou portàtil	42,31 €/ mes	4 mesos	169,24 €	6,4	10,83 €
Total					559,94 €

Taula 6 Costos dels imprevistos

6.2.5 Pressupost

Seguidament, mostrarem una taula que recull el pressupost final tenint en compte el pressupost inicial i totes les desviacions possibles que he hem tingut presents.

Producte	Cost
Costos directes	12.153,93 €
Costos indirectes	677,42 €
Contingència	1.924,70 €
Imprevistos	559,94 €
Total	15.315,99 €

Taula 7 Cost total

7 Sostenibilitat i viabilitat del projecte

7.1 Econòmica

Durant la gestió econòmica, també s'ha de tindre en compte el cost d'allotjar l'API en un servidor web. Inicialment, el que es farà, és allotjar-la en un local de la màquina, després, es pot decidir allotjar a un servidor de GESSI, si així ho troba convenient. Si decideix allotjar-ho en un servidor extern per temes de disponibilitat i seguretat que donen en alguns llocs dedicats, tindrà uns altres costos, però també es pot fer.

A més cal dir, que el servidor on pugui estar allotjada (dins de GESSI) i, en el futur, pugui estar-hi, si es decideix traslladar-lo, no és un servidor sencer dedicat a tenir-la. Per tant, el cost és difícil d'estimar.

De moment, el cost és zero, doncs no estem en la fase final de producció i el desplegament del nostre servei es fa en un servidor local.

7.2 Social

Actualment, la dimensió social del lloc on vivim, Catalunya, és d'incertesa i agitació a nivell polític i ciutadà. Pot ser un moment històric. A diferència del que succeïa anteriorment, ara tothom pot informar dels fets i la seva opinió al respecte.

Per tot això, aquesta eina pot ser molt important, per saber tot el que sent la població al respecte i contrastar-ho amb els discursos polítics. Pot ser una eina que serveixi de criteri per saber de primera mà la realitat del que succeeix. D'aquesta manera, actuar en conseqüència. És un pas més enllà, del qual es pot fer servir l'eina implementada per nodrir-se de les dades necessàries.

Les xarxes socials han estat un gran avenç per saber que passa a tot arreu. Però hi ha vessants per tots gustos, així que és difícil saber quina és la correcta. Amb aquesta eina, es podran desenvolupar solucions que permetin esbrinar-ho.

Un col·lectiu que podria veure's afectat pel nostre projecte és el dels mitjans de comunicació. Ara transmeten allò que els interessa, però es podrà crear una aplicació que faci neutra aquesta situació. De la mateixa manera, també els polítics es poden veure afectats, i mandataris d'arreu del món. Doncs, és ben sabut, que qui controla la informació, té el poder.

7.3 Ambiental

CO₂ emès per la generació d'electricitat durant el desenvolupament del projecte:

$$43,09 \text{ kWh} \times 308 \text{ g CO}_2/\text{kWh} = 13271,72 \text{ g CO}_2^3$$

Després de la realització del projecte, dependrà del servidor on s'allotgi, que serà compartit.

Es podrien suprimir les impressions de la memòria del TFG, però per recomanació de professorat de la universitat, s'ha dit als alumnes que el tribunal de la defensa sempre les prefereix.

Les emissions de CO₂ produïdes durant la realització del projecte, són inevitables. En canvi, durant la vida útil de l'eina, podem minimitzar el consum elèctric i les emissions de CO₂ mitjançant una programació eficient.

S'ha intentat treballar amb productes d'empreses que estan compromeses amb el medi ambient, com Apple®⁴ o el monitor de Dell⁵ que utilitzem, que utilitza un 25% de plàstics reciclats en el cos del producte i no usa arseni en la pantalla.

3

http://canviclimatic.gencat.cat/es/reduex_emissions/factors_demissio_associats_a_l_energia/

⁴ <https://www.apple.com/es/environment/>

5

http://accessories.euro.dell.com/sna/productdetail.aspx?c=es&l=es&s=dhs&cs=esdhs_1&sku=210-ADUL

8 Sistema de monitoratge

En aquest punt del document, ja hem vist diferents aspectes generals i comuns a tot projecte. Tot seguit, concretem sobre el disseny, funcionalitat i implementació del projecte que ens ocupa.

És important recalcar, que contràriament als serveis que proporcionava l'antiga API oficial d'Instagram, la nova ha estat desenvolupada per Facebook i ha tancat moltes possibilitats. Per poder fer ús de la nova API d'Instagram, s'ha de tenir un compte d'empresa, desenvolupar el servei que hem implementat, i aleshores, demanar a Facebook que ens concedeixin tenir l'API de manera real a la plataforma d'Instagram. Ara només podem fer servir l'API que hem desenvolupat de manera tancada. Hem d'agafar manualment les dades que ens interessin, i utilitzar-les com a entrada cap a la nostra API.

Degut a això, no podem fer una monitorització en temps real, si no, que més aviat hem hagut d'implementar un simulador que generi els missatges que envia l'API oficial d'Instagram (*Webhooks*) cap a la API que hem implementat (objectiu d'aquest projecte). A "*grosso modo*", quan algú escriu un missatge en alguna de les imatges que has publicat a la xarxa social d'Instagram, la API d'Instagram agafa l'identificador d'aquell comentari i l'envia a la nostra API amb un format determinat i amb altres camps. Com que Instagram només ens dóna el format de les peticions que ens enviarà en un futur quan l'API que implementem estigui aprovada, som nosaltres els que hem hagut de simular el comportament de l'API d'Instagram.

Ja hem comentat que aquest punt s'escapa dels objectius del projecte, així que l'hem simulat de manera senzilla sense tindre en compte el procés fet en temps real ni l'espai temporal en que es realitzen els enviaments d'aquests avisos.

8.1 Monitors

En l'àmbit d'aquest projecte, la monitorització és el fet de tenir quelcom que necessitem o desitgem que sigui comprovat durant un cert temps i amb una freqüència determinada, disponible per a ser obtingut.

Per a cada instància que es monitoritzi, s'executarà un procés de monitoratge constituït per dues tasques. Aquest procés de monitoratge i les seves tasques són tals degut a que

el monitor d'Instagram forma part d'un sistema de monitoratge que consta per diversos monitors, i han de ser heterogenis per facilitar l'execució paral·lela de tots ells i el seu funcionament sota un mateix sistema. Les tasques o operacions es corresponen amb els següents requisits funcionals:

- Inicialitzar el procés de monitoratge: s'inicia el procés definint els diferents paràmetres necessaris per a que es realitzi el procés.
- Aturar el procés de monitoratge: en qualsevol moment es pot decidir que es pari el procés de monitoratge degut a qualsevol decisió pertinent que es prengui o necessitat determinada.

Un cop s'aturi el procés de monitorització, els resultats hauran d'estar disponibles a algun lloc per tal de que es puguin consultar, però mai estaran disponibles dins del servidor de manera indefinida.

8.1.1 Especificacions tècniques i funcionals

El nostre sistema és una eina *open source* i volem que pugui ser usat com a mòdul d'altres sistemes o ampliat. Per tal de fer possible aquesta futura *feature*, el nostre sistema ha de ser heterogeni (flexible) i compatible amb molts possibles consumidors, actuals o futurs.

Abans de parlar de l'arquitectura del sistema, es considera que serà útil tenir els següents coneixements sobre el monitor d'Instagram que hem implementat:

- **Comportament intern del monitor:** saber com actua el monitor a nivell intern a partir d'analitzar les seves necessitats i de tenir més coneixements tècnics sobre el coneixement del propi sistema.
- **Ús de les dades obtingudes pel monitor:** saber quin és el recorregut de les dades i quines facilitats se li donen a l'usuari per a que pugui fer-ne ús i veure-les, tot respectant la seva privacitat.
- **Configuració del monitor:** s'indicaran quins són els paràmetres necessaris per a que el monitor pugui exercir la seva funcionalitat.

8.1.1.1 *Comportament intern del monitor*

Un monitor és un component software que s'encarrega de comprovar l'estat d'un altre component software amb una freqüència de temps determinada (o constantment).

En el cas específic que ocupa aquest projecte, es comprova l'estat del que ens envia el component de software del simulador (d'Instagram, que és l'encarregat d'enviar la *notificació* cada vegada que hi ha dades noves) amb una freqüència de temps determinada i escollida per l'usuari. Aquesta freqüència de temps pot ser constant o aleatòria, depenent del que se l'indiqui al simulador. Li podem indicar al simulador que generi els missatges que ha d'enviar al monitor amb un temps constant determinat, o amb un temps aleatori comprès entre dues unitats de temps concretes.

A continuació veurem les dades que es necessiten per tal de poder començar el procés de monitorització i explicarem per què.

Per facilitar la seva comprensió, les reunirem semànticament en dos grups diferents, *Elements per la monitorització* (que s'encarrega de configurar el monitor per tal de que sigui possible realitzar el procés) i *Elements per la generació de missatges* (que s'encarrega de realitzar la generació de missatges amb destí al monitor).

- Elements per la monitorització:
 - Access Token: identifica el nostre usuari amb aquesta clau per tal de permetre'ns fer peticions durant un temps determinat. És molt important tenir aquesta clau de manera segura i passar-la al nostre monitor també de forma segura, altrament, pot ser interceptada i en mans de segons qui, és com donar la contrasenya del nostre compte d'usuari de la xarxa social (però durant un cert temps determinat).
 - Paraula clau (o keyword): necessària quan volem monitoritzar elements (comentaris o mencions) que tinguin dins del seu text la paraula determinada.
- Elements per la generació de missatges:
 - Fitxer per a la generació de missatges de comentaris: es tracta d'un fitxer de text que conté totes les dades dels identificadors de comentaris que ens passaria la Graph API d'Instagram. Aquest fitxer en conté diverses, però el fet és que, Instagram rep un element cada vegada que es produeix un comentari sobre una de les nostres publicacions. Aquesta ha estat la

manera de solucionar la impossibilitat de fer una simulació del que faria la xarxa social amb la seva API (perquè no ens han acceptat la nostra aplicació, encara, doncs estem en un procés de desenvolupament i no hem passat a producció encara).

- Unitats de temps per la generació de missatges de comentaris en temps constant: indiquem al simulador que generi un nou missatge de comentari cada n unitats de temps.
- Unitat de mesura per la generació de missatges de comentaris en temps constant: lligat amb l'anterior, li diem que la mesura de cada unitat de temps (donades anteriorment) és segons, minuts o hores.
- Rang d'unitats de temps per la generació de missatges de comentaris en temps aleatori: donem dues unitats de temps que formen un interval (la 1a. menor que la 2a.). D'aquesta manera, es generarà un nou missatge de que hi ha un nou comentari amb una unitat de temps compresa entre la primera i la segona unitat de temps.
- Unitat de mesura per la generació de missatges de comentaris en temps aleatori: lligat amb l'anterior, li diem que la mesura de cada unitat de temps (donades anteriorment) és segons, minuts o hores.
- Fitxer per a la generació de missatges de mencions: fitxer que conté les dades dels missatges que s'enviaran al monitor cada vegada que el simulador hagi de enviar una notificació al monitor que hi ha una nova menció i les dades que identifiquen la menció corresponent.
- Unitats de temps per la generació de missatges de mencions en temps constant: indiquem al simulador que generi un nou missatge de menció cada n unitats de temps.
- Unitat de mesura per la generació de missatges de mencions en temps constant: lligat amb l'anterior, li diem que la mesura de cada unitat de temps (donades anteriorment) és segons, minuts o hores.
- Rang d'unitats de temps per la generació de missatges de mencions en temps aleatori: donem dues unitats de temps que formen un interval (la 1a. menor que la 2a.). D'aquesta manera, es generarà un nou missatge

de que hi ha una nova menció amb una unitat de temps compresa entre la primera i la segona unitat de temps.

- Unitat de mesura per la generació de missatges de mencions en temps aleatori: lligat amb l'anterior, li diem que la mesura de cada unitat de temps (donades anteriorment) és segons, minuts o hores.

En el front-end es comprova mitjançant el JavaScript que les dades del formulari compleixen els requisits, com ara:

- L'Access Token té una longitud suficient, no és buit i és vàlid segons la Grapn API d'Instagram.
- S'ha seleccionat almenys un fitxer per comentaris o per mencions.
- Els valors dels temps són correctes.

Aleshores, es fan les peticions GET / POST per pujar les dades dels formularis al servidor. Al back-end es faran altres comprovacions, com ara, que el contingut dels fitxers que s'han pujat al servidor són vàlids.

Aleshores, el simulador començarà a enviar peticions cap al monitor, tot complint amb els valors de temps específics.

Quan el monitor rep una petició amb el JSON que conté la identificació del comentari o menció que s'ha fet, processa la petició per obtenir l'identificador i poder fer la petició demanant els recursos específics que s'han cregut oportuns i poder-los mostrar per pantalla al front-end i guardar-los en el fitxer de text mentre la instància de monitorització està activa o ja no ho està però no s'ha iniciat cap altre petició per comentaris o mencions, segons escaigui.

Les dades que ens envia el monitor per poder fer la monitorització de comentaris són:

```
{
  "field": "comments",
  "value": {
    "id": "18058826815152799",
    "text": "Looks like we've more calorous day 🥰"
  }
}
```

Fig 5 Exemple JSON del recurs "comentaris" del Simulador

- fields: indica si el recurs que s'envia al monitor és un comentari ("comments") o una menció ("mentions").

- value: JSON que conté informació útil per poder monitoritzar el recurs o possiblement útil.
- id: identificador del comentari, es pot utilitzar per trobar informació de manera ràpida dins de la Graph API d'Instagram.
- text: contingut de text del comentari.

Degut a que aquest JSON difereix amb el del recurs de mencions, trobem interessant posar un exemple del JSON que s'usa per part del simulador per enviar-lo al monitor. Com que n'hi ha de comuns, només posarem els que difereixen.

```
{
  "field": "mentions",
  "value": {
    "media_id": "17864280508406328",
    "comment_id": "17890551145342490"
  }
}
```

Fig 6 Exemple JSON del recurs "mencions" del Simulador

- media_id: identificador de la imatge o fotografia que s'ha pujat a la xarxa social i sobre el quals es fa un comentari amb la menció de l'usuari (del que fa el procés de monitorització). Es poden buscar diversos elements relacionats amb la fotografia a partir d'aquest camp.
- comment_id: identificador del comentari on ens han mencionat. És equivalent al camp "id" dels comentaris (Fig 1).

A nivell informatiu, cal esmentar que una menció és exactament igual que un comentari, però en la menció se'ns menciona el nostre usuari (@jinstaapio). Per tant, quan es monitoritzen comentaris i mencions, és possible que un recurs aparegui en els dos llocs per separat.

Un cop que el monitor rep algun d'aquests JSON enviats pel Simulador, processa el JSON amb l'objectiu d'obtenir l'identificador del comentari (o menció) i fer una petició a la Graph API d'Instagram per tal d'obtenir les dades desitjades.

Les dades que es reben després de fer la petició són les que es mostren a continuació:

```
{
  "id": "17867026186459652",
  "text": "Woooooooooooooooooooooooooooo",
  "user": {
    "id": "17841406152924021"
  },
  "username": "jinstaapio",
  "like_count": 0,
  "timestamp": "2019-08-21T15:17:22+0000",
  "media": {
    "id": "17983871017269320"
  }
}
```

Fig 7 JSON amb les dades d'un element de la monitorització

- id: identificador del comentari (o comentari que conté la menció).
- text: text que conté el comentari.
- user: identificador de l'usuari que ha fet el comentari o menció a la publicació del compte d'Instagram monitoritzat.
- username: nom de l'usuari que ha fet el comentari / menció.
- like_count: número de *M'agrada* que té el comentari o comentari que conté la menció.
- timestamp: data i hora en que s'ha fet el comentari / menció.
- media: identificador de la fotografia que s'ha pujat i que conté el comentari / menció concrets.

8.1.1.2 Ús de les dades obtingudes pel monitor

En aquest subapartat especificarem de forma concreta el que s'ha deixat entreveure en el subapartat anterior ([8.1.1.1 Comportament intern del monitor](#)).

Així doncs, ara veurem el flux de les dades entre component i component. A continuació, amb la següent figura:

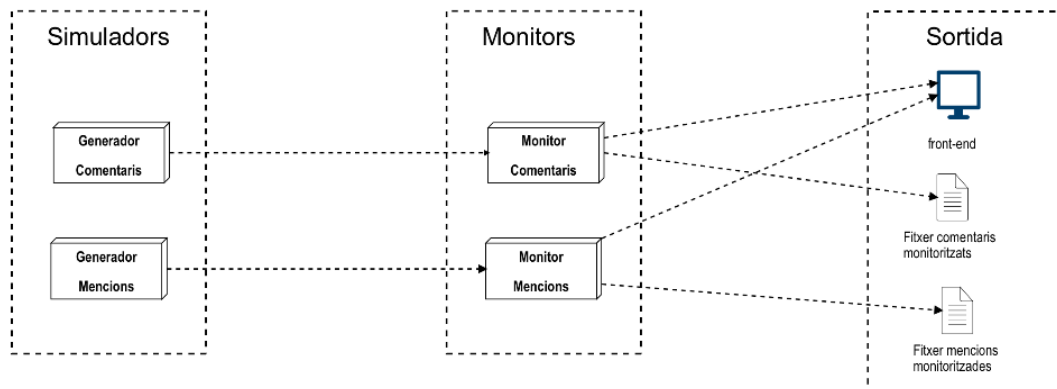


Fig 8 Ús de les dades

Com podem veure, veiem un diagrama que mostra el flux de les dades del programa, tal i com s'ha dit anteriorment.

No incloem la obtenció de dades que fa cadascun dels monitors fent peticions contra la Graph API d'Instagram, ja s'entén que és una tasca que fa el nostre monitor.

Un cop configurats els diferents components del sistema (Simuladors per la generació de missatges i els monitors), els simuladors enviaran els missatges cap als monitors pertinents.

Aleshores, el o els monitors, obtindran les dades mitjançant les peticions a la Graph API d'Instagram i de les entrades que es van rebent amb els simuladors com origen. Les dades rebudes des dels simuladors, són necessàries per fer la petició a la Graph API d'Instagram. L'adreça web d'aquesta Graph API és <https://graph.facebook.com>.

Un cop que el/s monitor/s del nostre sistema ja ha obtingut les dades que necessita, les desa en un o dos fitxers, segons els recursos que s'estan monitoritzant i el monitor que ho fa (Monitor de comentaris o Monitor de mencions). A més, perquè l'usuari pugui visualitzar els elements monitoritzats en temps real, els resultats també es mostren per la interfície web.

8.1.1.3 Configuració del monitor

Per a que el procés de monitorització es pugui realitzar cal que certs paràmetres del simulador i del monitor estiguin presents. Els paràmetres que el configuren apareixen as subapartat anterior ([8.1.1.1 Comportament intern del monitor](#)) agrupats en dos sectors, *Elements per la monitorització* | *Elements per la generació de missatges* – del simulador).

En quant a *Elements de la monitorització* recordem que tenim:

- Access Token.
- Paraula clau (o keyword).

Mentre que pels *Elements per la generació de missatges* tenim:

- Fitxer per a la generació de missatges de comentaris.
- Unitats de temps per la generació de missatges de comentaris en temps constant.
- Unitat de mesura per la generació de missatges de comentaris en temps constant.
- Rang d'unitats de temps per la generació de missatges de comentaris en temps aleatori.
- Unitat de mesura per la generació de missatges de comentaris en temps aleatori.
- Fitxer per a la generació de missatges de mencions.
- Unitats de temps per la generació de missatges de mencions en temps constant.
- Unitat de mesura per la generació de missatges de mencions en temps constant.
- Rang d'unitats de temps per la generació de missatges de mencions en temps aleatori.
- Unitat de mesura per la generació de missatges de mencions en temps aleatori.

Parlant purament de l'objectiu que ens ocupa en aquest TFG, ens centrarem amb l'Access Token i la paraula clau.

- Access Token: identifica de manera única al nostre usuari per tal que puguem realitzar les peticions contra la Graph API d'Instagram.

En realitat, tal com es pot deduir de la URI donada abans, és la Graph API de Facebook, doncs Instagram pertany a Facebook. Però gràcies a la configuració de l'aplicació que hem fet a la web de desenvolupadors de Facebook, el nostre Access Token només tindrà accés a poder fer peticions sobre el nostre compte d'Instagram, així com el fet que només podrà fer les consultes necessàries per obtenir les dades que nosaltres desitgem. Els permisos justos i correctes que volem són necessaris, doncs no volem donar permisos per a que ens esborrin els comentaris o les mencions que tenim, o altres múltiples possibilitats.

- Paraula clau: aquest camp és opcional i serveix per indicar-li al nostre monitor que només volem que es tinguin en compte els elements que en el seu camp “text” tinguin entre les seves paraules, la paraula indicada. Es pot veure la imatge on surt la estructura del JSON que genera el monitor i on hi apareix el camp “text”. [Cliqueu per anar a la figura.](#)

8.1.2 Disseny i arquitectura genèrica

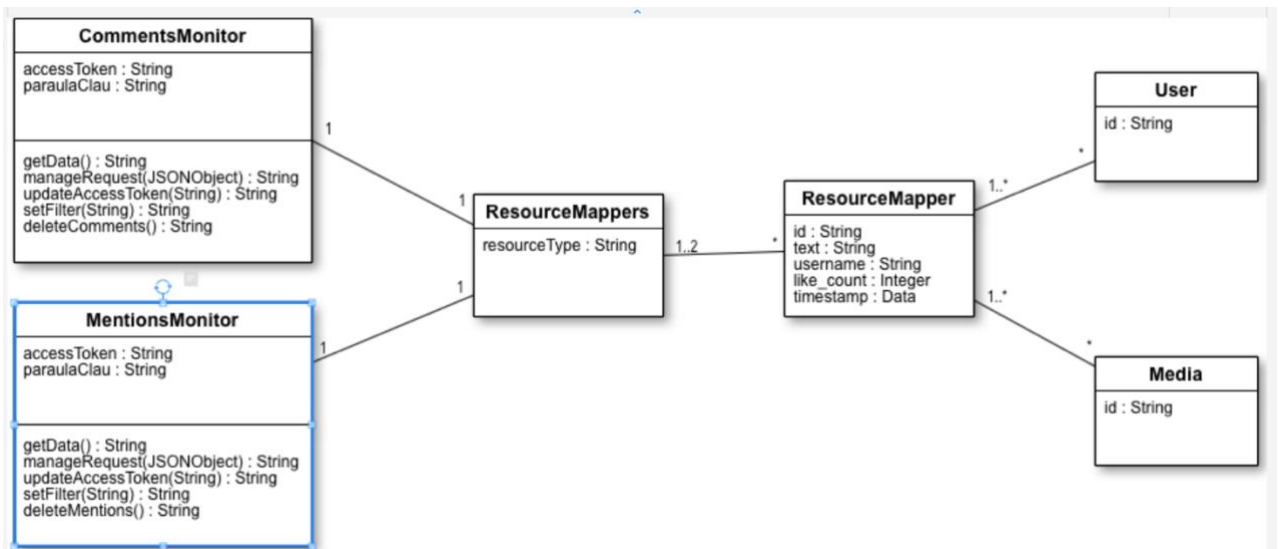


Fig 9 Diagrama de les classes Java dels monitors

En aquesta figura es pot observar l’estructura del diagrama de classes (Java) dels monitors del sistema que hem implementat.

Observem una peculiaritat que ara mateix justificarem. Les dues classes dels monitors, *CommentsMonitor* i *MentionsMonitor*, són molt semblants en quant a atributs i funcionalitats fa referència. Podríem haver aplicat polimorfisme i usar un patró conegut que ens aporta molts avantatges, com ara la facilitat alhora de mantenir el software del sistema.

Tot i això, l’inconvenient que ens ha fet renunciar a aquesta qualitat del software, és el fet de que incrementàriem el temps que el sistema triga per obtenir els resultats. Degut a que les xarxes socials i el monitoratge són quelcom que transcorre en viu, en temps real, no volem perjudicar el temps. Si donem menor prioritat al temps, l’experiència de l’usuari es veu afectada i volem que el nostre producte premii les qualitats que em raonat que ens convenen pel tipus de producte que hem implementat.

Una vegada que hem esclarit aquesta qüestió, descrivim les classes, atributs i mètodes que menys trivials creiem que són a l'hora d'intuir el seu significat:

- **CommentsMonitor**: classe que s'encarrega de recollir els identificadors (entre altres) de tots els comentaris que s'han produït “en temps real”, obtenir totes les dades a partir de l'identificador i mostrar-les a l'usuari (així com guardar-les temporalment en un fitxer). A continuació mostrem els dos paràmetres que necessitem per poder obtenir les dades que volem tindre a partir de l'identificador del comentari i els mètodes que tenim per comunicar-nos amb el front-end i la Graph API oficial d'Instagram.
 - **accessToken**: token necessari per poder-li fer les peticions a la Graph API oficial d'Instagram. Amb aquest, ens identifiquem de cara a la Graph API, té validesa durant un període de temps determinat i és introduïda per l'usuari en el front-end.
 - **paraulaClau**: String que serveix per filtrar els comentaris que rebem i obtenir només aquells que tenen en el seu text aquest determinat String.
 - **getData()**: funció que es crida des del front-end per obtenir tots els comentaris que s'han monitoritzat fins el moment.
 - **manageRequest(JSONObject)**: funció que es crida des del simulador de comentaris per indicar-li al monitor de comentaris que hi ha un nou comentari i quines són les seves dades bàsiques. Aquestes dades estan incloses dins del JSONObject que se li passa com a paràmetre d'entrada. Podem veure un exemple de com és un d'aquests paràmetres en [aquesta figura anterior](#).
 - **updateToken(String)**: funció que es crida des del front-end per assignar el valor de l'accessToken al monitor. El valor del paràmetre que es passa per l'entrada és el de l'accés token. Òbviament, per major seguretat, aquest paràmetre es passa codificat dins de la URI.
 - **setFilter(String)**: funció que es crida des del front-end per assignar el valor de l'atribut paraula Clau de la classe. El valor del paràmetre d'entrada és introduït per l'usuari en la interfície gràfica.
 - **deleteComments()**: funció que és cridada pel front-end un cop que es comença un nou procés de monitorització (o s'apaga/reinicia el servidor),

per tal que no quedi rastre de les dades obtingudes i assegurar la privacitat de l'usuari.

Aquesta i les següents classes, defineixen l'estructura del JSON resultant dels elements monitoritzats. També s'utilitzen com a objectes que pertanyen a les classes *CommentsMonitor* i *MentionsMonitor*.

- ResourceMappers: classe que conté l'estructura dels comentaris monitoritzats.
 - resourceType: indica si el conjunt de recursos monitoritzats són comentaris (*comments*) o mencions (*mentions*).
- ResourceMapper: classe que recull l'estructura d'un comentari (o menció) monitoritzat.
 - id: identificador del comentari monitoritzat. El monitor accedeix a aquest mitjançant la crida:
http://grap.facebook.com/id?fields=field1,field2,...,fieldN&access_token=accessToken
Sent *field1* , ..., *fieldN* el nom de cadascun dels camps que es sol·liciten a la Graph API oficial d'Instagram (com els d'avall, amb el nom que sigui correcte), i *accessToken* és el valor de l'accés token, guardat a la classe del monitor i que ha introduït l'usuari.
 - text: és el contingut que s'ha escrit com a comentari.
 - username: nom de l'usuari que ha escrit el comentari. És de l'estil *@username*.
 - like_count: número de *M'agrada* que ha rebut el comentari.
 - timestamp: data en que s'ha fet el comentari.
- User: classe que identifica el usuari ha fet el comentari.
 - id: identificador únic de l'usuari que ha fet el comentari. És un conjunt de números.
- Media: classe que identifica la fotografia / vídeo que es comenta.
 - id: identificador de la fotografia / vídeo que es comenta.

Altres mètodes:

Cal aclarir que, a part dels mètodes que s'han mencionat i mostrat aquí, n'hi ha d'altres, com ara *getters* i *setters* que no mostrem perquè ens sembla superflu i recarregaríem innecessàriament el contingut del document.

També hem sobreescrit mètodes de tipus *toString()* per retornar el contingut de les classes que hem fet servir per definir les estructures dels JSON.

A més, hem sobreescrit mètodes *objecte.equals(Object)* per majors comoditat i practicitat.

Per acabar, hem fet servir llibreries de Jackson per fer el mapeig entre objectes JSON i els objectes que representen.

8.1.3 Implementació de monitors

En aquesta secció, mostrarem alguns aspectes de la implementació del monitor de la xarxa social d'Instagram.

Instagram Monitor

L'objectiu d'aquest monitor és recol·lectar les diferents publicacions de la xarxa social Instagram on se'ns ha esmentat o veure els comentaris i "*M'agrada*" que s'han fet a les nostres pròpies publicacions.

Per complir amb aquest objectiu, fem servir l'API proporcionada per *Facebook, Instagram Graph API*.

Els atributs interns que es necessitaran a la nostra API són els següents:

Exposició com a servei REST

Els monitors que hem realitzat ofereixen serveis REST i ara mostrarem les capçaleres de les funcions, així com justificarem algunes de les decisions preses.

Els dos monitors són molt semblants, així que només ensenyarem les capçaleres de les funcions de la classe CommentsMonitor que justifiquen que es dona un servei REST.

```
@Path("/comments")
public class Comments {

    @POST
    @Path("/id")
    @Consumes(MediaType.APPLICATION_JSON)
    @Produces(MediaType.TEXT_PLAIN)
    public String manageRequest(InputStream commentsInputStream)
    { ... }

    @POST
    @Path("/update/token")
    @Consumes(MediaType.APPLICATION_FORM_URLENCODED)
    @Produces(MediaType.TEXT_PLAIN)
    public String updateAccessToken(@FormParam("access_token")
String accessToken) { ... }

    @GET
    @Path("/get")
    @Produces(MediaType.TEXT_PLAIN)
    public String getData() { }

    @GET
    @Path("/filter/{word}")
    @Produces(MediaType.TEXT_PLAIN)
    public String setFilter(@PathParam("word") String filterWord)
    { ... }

    @DELETE
    @Path("/delete")
    @Produces(MediaType.TEXT_PLAIN)
    public String deleteComments() { ... }

}
```

La primera línia vol dir que per fer qualsevol petició dins d'aquest projecte / monitor, s'haurà de fer amb la següent URI:

[http://localhost:8080/MonitorInstagram/rest/comments/...](http://localhost:8080/MonitorInstagram/rest/comments/)

El següent mètode, *manageRequest(InputStream)* és cridat des del simulador, que genera missatges amb la estructura adequada per ser de tipus comentari, tal i com veiem a [l'anterior figura Fig 3](#) i s'encarrega de fer totes les gestions per obtenir les dades del comentari específic fent les crides necessàries a la Graph API oficial d'Instagram,

guardar les dades de totes les dades obtingudes fins al moment en un fitxer de text i en estructures internes de la classe per tal de que estiguin disponibles en el moment específic en que se li demanin.

S'utilitza un mètode POST perquè l'entrada que se li passa no és factible de ser passada per un mètode GET i perquè es fan modificacions a la classe i als seus atributs. Així que sintàcticament, té tota la lògica.

El següent mètode és *updateAccessToken(String)* i s'utilitza per actualitzar o donar valor a l'*accessToken* que tenim dins de la classe *CommentsMonitor*. Es fa servir un mètode POST per major seguretat de les dades sensibles que es passen quan es fa la petició. El tipus de dades que es consumeix quan es fa la petició, l'*String* paràmetre d'entrada *accessToken*, també es passa amb tipus d'entrada consumida *MediaType.APPLICATION_FORM_URLENCODED* per evitar que sigui una dada visible quan es veuen les dades passades amb el POST i afegir-hi un nivell de seguretat extra.

El següent mètode, *getData()*, retorna les dades de tots els comentaris que s'han monitoritzat fins el moment.

Continuem amb el mètode *setFilter(String)*, que serveix per indicar-li al monitor quina és la *paraulaClau*, imprescindible per saber la paraula que ha d'aparèixer en els textos dels comentaris que es monitoritzen. Com a peculiaritat, per fer una acció d'aquest tipus, normalment es faria servir un mètode de tipus POST, però hem decidit usar-ne un de tipus GET per termes d'eficiència. Els mètodes POST són més costosos en temps i més segurs per les dades que es passen. Però hem arribat a la conclusió que la *paraula Clau* no és una dada sensible, és un tipus de dada curta (una paraula) i per tant, podem passar-la en un mètode GET com a paràmetre de la URI.

Per acabar, tenim la funció *deleteComments()*, que s'encarrega de netejar totes les dades relatives a la monitorització executada. Aquest mètode s'executa quan acaba el procés de monitorització i n'ha de començar un de nou. Seguint la lògica dels tipus de mètodes REST, hem fet servir una crida amb un mètode de tipus DELETE. El resultat retornat és un text (tal com es mostra a la declaració de la funció i al *@Produces*), que indica si la operació s'ha pogut realitzar amb èxit.

8.2 Tecnologies utilitzades

A continuació mencionarem, descriurem i justificarem les tecnologies que s'usen al projecte.

- Gradle: sistema d'automatització que ens facilita descarregar / importar llibreries necessàries pel desenvolupament del nostre projecte, així com gestionar les dependències i l'ordre d'execució dels components del projecte. A més, permet una gran escalabilitat en sistemes grans, compostos de múltiples paquets.
- HTML5: llenguatge de programació i estàndard per fer pàgines web, basat en etiquetes que doten de sintaxis a les diferents seccions que componen la mateixa pàgina web. Aquestes etiquetes són determinades per l'estàndard. HTML5 s'utilitza tot combinant amb JavaScript per tal d'interactuar amb l'usuari i fer de pont amb el back-end de la nostra API.
- Java: llenguatge de programació. L'utilitzem degut al gran nombre de llibreries disponibles per aquest, fet que ens facilita la programació a partir de certa base, sense necessitar reinventar la roda.
- JavaScript: llenguatge lleuger i interpretat, orientat a objectes, més conegut com a llenguatge script per a pàgines web (també s'utilitza en altres entorns sense navegador web, com node.js). L'utilitzem perquè és molt lleuger, de tal manera que dota al nostre servei de velocitat. L'usuari tindrà la resposta a la seva petició ràpidament.
- JSON: *JavaScript Object Notation*, dades estructurades generades per la API RESTful. Podríem haver utilitzat XML en comptes de JSON, però ens hem decantat per aquest degut a que és millor en aquests aspectes:
 - Més senzill fer-ne un anàlisi gramatical (*parse*).
 - És traduït en un objecte JavaScript preparat per a utilitzar, mentre que l'XML no.
 - És més curt i entenedor.
 - Permet l'ús d'*arrays* o vectors.
- RESTful: Servei web que funciona amb una API amb l'objectiu de servir peticions als seus clients. Aquestes peticions, es fan amb un mètode HTTP i el seu resultat

serà servit en un JSON al client interessat (acostuma a ser un programa o una API que consumeix aquestes dades). Quan les respostes generades per la API han sigut simples, per exemple, un “Ok”, hem preferit fer servir el format de simple text, doncs no cal fer servir un format que està pensat per simplificar respostes llargues i/o complexes, tot millorant la seva comprensió.

- Web-technologies: tecnologies web que permeten el desenvolupament d'aplicacions web amb certa facilitat, ja que disposen d'un ventall ampli de llibreries que en faciliten la programació. D'aquesta manera, utilitzem solucions existents, implementades i àmpliament provades, tenint la certesa de que funciona bé i eficientment.

9 Interacció amb el sistema

Com a part visual del nostre projecte, volem desenvolupar una petita interfície web que mostri a l'usuari el potencial del sistema. Volem mostrar què pot aportar l'API que hem desenvolupat a l'hora de que tercers decideixin utilitzar el producte resultant del projecte.

9.1 Anàlisi de requisits

Els requisits que cal que es satisfessin en aquesta part són:

- Validació de l'Access Token introduït per l'usuari.
- Inserir una paraula clau.
- Inserir els elements necessaris per a la generació de missatges del simulador: fitxers de text amb l'entrada, freqüències de generació de missatges (constants o aleatòries).
- Mostrar les dades monitoritzades.
- Donar resposta guiada i intuïtiva sobre l'estat del procés a l'usuari.

9.2 Disseny de la interfície

Ara mostrarem l'estructura que té la interfície gràfica, mitjançant la qual, l'usuari pot interactuar amb el sistema.

Hem dividit l'espai en tres zones, que a la vegada, es poden dividir en dues zones segons la seva semàntica.

The image shows a web interface titled "Monitorització d'Instagram". It is divided into three main sections:

- Elements per la monitorització:** Contains a "Token:" input field and a "Filtrar per paraula:" input field.
- Elements de la generació de missatges:** Contains a section "Selecció dels fitxers de generació de missatges:" with a dropdown menu showing "Seleccionar archivo" and "Ningún archivo seleccionado". Below this are two radio button options: "Temps de generació constant:" (selected) and "Temps de generació aleatòria:". Each option has a numeric input field (set to 1) and a unit dropdown menu (set to "segons").
- Mencions:** Contains a dropdown menu showing "Seleccionar archivo" and "Ningún archivo seleccionado", followed by a "Temps de generació constant:" radio button (selected) and a numeric input field (set to 1) with a unit dropdown menu (set to "segons").

At the bottom of the interface, there are two buttons: "Cancel" and "Monitor".

Fig 10 Estructura interfície gràfica

A nivell semàntic, diferenciem la parts esquerra i dreta.

A l'esquerra tenim les entrades que fa l'usuari per tal de que els monitors i simuladors estiguin configurats.

A la dreta tenim els resultats de la monitorització (comentaris a dalt, mencions a baix). La part esquerra, que correspon a les configuracions esmentades, la dividim per la configuració dels monitors (quadrat superior) i la dels simuladors que generen els missatges (quadrat inferior inferior).

Ara entrarem més en detall en cadascun dels elements de les tres parts:

- **Token:** és l'accés token dels dos monitors, necessari per poder fer les consultes a la graph API oficial d'Instagram.
- **Filtrar per paraula:** és l'atribut *paraulaClau* que, igual que el Token, ja hem vist anteriorment (veure la [figura 7](#)). És un paràmetre opcional que omplirem en cas que vulguem que els comentaris / mencions monitoritzats tinguin dins del seu element "text" la paraula escrita.
- **Selecció del fitxer de Comentaris:** en cas que volem monitoritzar comentaris, hem d'indicar un fitxer de text vàlid per la generació de missatges per part del simulador. Si el fitxer no és vàlid, el programa no funcionarà adequadament.
- **Selecció de temps constant (comentaris):** si el *radio* està marcat i volem monitoritzar comentaris (perquè ja hem seleccionat el fitxer o el volem seleccionar més endavant), aquí li indiquem el temps de generació de missatges del simulador en temps constant. Haurem d'introduir la quantitat de temps i l'escala de mesura de temps (segons, minuts, hores). El significat de constant és que el missatge es generarà cada *n* segons, minuts o hores. Aquest temps pot variar lleugerament degut que s'ha d'inicialitzar el simulador i la primera vegada que es genera un missatge pot trigar més degut a aquest motiu.
- **Selecció de temps aleatori (comentaris):** funciona igual que l'anterior, però en aquest cas, volem que el missatge que genera el simulador, es generi en un temps comprès entre els dos valors indicats més endavant (per defecte, entre 1 i 2 segons). El primer valor ha de ser menor que el segon, en cas erroni, se'ns indicarà en el moment que cliquem el botó de començar la monitorització.
- **Selecció del fitxer de Mencions:** en cas que volem monitoritzar mencions, hem d'indicar un fitxer de text vàlid per la generació de missatges per part del simulador. Si el fitxer no és vàlid, el programa no funcionarà adequadament.

- **Selecció de temps constant (mencions):** si el *radio* està marcat i volem monitoritzar mencions (perquè ja hem seleccionat el fitxer o el volem seleccionar més endavant), aquí li indiquem el temps de generació de missatges del simulador en temps constant. Haurem d'introduir la quantitat de temps i l'escala de mesura de temps (segons, minuts, hores). El significat de constant és que el missatge es generarà cada *n* segons, minuts o hores. Aquest temps pot variar lleugerament degut que s'ha d'inicialitzar el simulador i la primera vegada que es genera un missatge pot trigar més degut a aquest motiu.
- **Selecció de temps aleatori (mencions):** funciona igual que l'anterior, però en aquest cas, volem que el missatge que genera el simulador, es generi en un temps comprès entre els dos valors indicats més endavant (per defecte, entre 1 i 2 segons). El primer valor ha de ser menor que el segon, en cas erroni, se'ns indicarà en el moment que cliquem el botó de començar la monitorització.

També hem tingut en compte en la part de la interacció i disseny d'interfícies per a que l'usuari es senti més còmode i tingui una millor experiència.

Respecte la monitorització, els fitxers pujats per la generació de missatges per part del simulador i els temps indicats, volem afegir més detalls sobre la implementació. En cas que no s'hagi pujat el fitxer d'un dels recursos, és indiferent els temps que se li posin, el front-end ho detecta. Addicionalment, si no es selecciona cap fitxer de text, se'ns mostra un missatge després d'haver premut el botó de *Monitor* (aquest terme és una abreviació de monitoritzar, fent servir el mateix terme però en anglès).

Quan es prem la zona indicada per pujar un fitxer, ens apareix un element emergent per poder seleccionar un fitxer local. Un cop el seleccionem i fem que aquell sigui l'escollit, l'arxiu es començarà a pujar al servidor. Si aquest s'ha pujat correctament, es llença un missatge (en color verd) i diu *Fitxer correcte!*.

Respecte a mostrar els elements obtinguts de la monitorització, van apareixent a mesura que es tenen disponibles a la zona corresponent, segons si es tracta de monitorització de Comentaris o Mencions. En cas de ser necessari, si la longitud del resultat ocupa la zona que es veu a simple vista, aleshores es pot fer *scroll* per visualitzar més contingut. L'espai del que disposem és reduït, volem mostrar tota la informació de la pàgina web en una sola plana. El contingut que es mostra és purament informatiu, ja

hem comentat que és una interfície senzilla, encara que volem que l'usuari es senti còmode.

Un cop ja hem inicialitzat la monitorització, no podem indicar-li una altra vegada a la interfície web que comenci un nou procés de monitoratge. Hem de seguir amb el mateix o prémer el botó de “Cancel·lar” per poder tornar a fer-ne un de nou.

Al prémer el botó de “Cancel·lar”, se'ns posen en blanc tots els camps i opcions que l'usuari ha d'omplir. D'aquesta manera, considerem que fem la interfície web més usable de cara a l'usuari.

També comentar, que quan hi ha un camp erròniament complimentat, es mostren els missatges d'error en color vermell, però el text és d'una mida més petita. Es vol ser una mica discret, perquè en termes d'experiència d'usuari, s'ha d'anar en compte, doncs podríem fer sentir violent a l'usuari.

9.3 Implementació

La implementació de la interfície gràfica s'ha fet amb el llenguatge HTML5 i JavaScript. Es va decidir al principi que es faria així, per una part, per desconeixement d'altres tecnologies, i per altra, perquè si ho haguéssim fet amb altres tecnologies, que a posteriori hem vist que serien millor pel temps de resposta amb l'usuari, s'hagués requerit més temps per aprendre les tecnologies i hi hagués hagut major probabilitat de fallida degut a un cúmul de desconeixements.

Perquè el tema sigui menys abstracte, concretaré. JavaScript és un llenguatge que s'estén en una important part del món de la programació web. És una tecnologia en la que es tenien nocions bàsiques i no es dominava del tot. Així que ens vam decantar per aquesta, doncs es podia fer el que volíem amb aquest llenguatge que complementa l'HTML5. Més endavant, ens hem trobat amb certs problemes de temps de resposta, i una bona solució era utilitzar nous *frameworks* que s'utilitzen amb JavaScript i/o *AJAX*, com ara *React*.

Concretant el que dèiem a nivell abstracte anteriorment, no coneixíem profundament JavaScript, no coneixíem React, i aquest framework requereix coneixements més amplis del llenguatge JavaScript dels que teníem. Per tant, és difícil de quantificar quant més temps s'hagués requerit per aprendre la tecnologia i els riscos posteriors associats a això.

10 Validació del sistema

10.1 Disseny de tests

Hem aprofitat la metodologia àgil utilitzada (vegeu [4.1 Introducció a KANBAN](#) per a més informació) per fer les proves requerides amb l'aplicació *Postman* (vegeu [5.2.2 Recursos materials](#), on s'anomena). S'ha anat comprovant que les crides funcionaven, així com els mètodes implicats que s'han implementat i que estaven involucrats amb la crida específica. D'aquesta manera, s'ha anat comprovant que el resultat d'allò que es feia era vàlid. Aquestes crides fetes amb *Postman* s'han guardat per poder-les utilitzar en qualsevol moment, molt útil durant la implementació i durant el funcionament posterior del sistema.

Fins a dia d'avui, es troba en un correcte funcionament.

10.2 Execució de tests

Tal com es deixa entreveure a l'apartat anterior, mentre s'ha fet la implementació del sistema, s'ha comprovat que els diferents mètodes i funcions que s'implementaven es trobaven en un correcte funcionament.

Executant els testos d'aquest mode ha estat positiu durant l'hora de desenvolupar el codi del sistema, doncs es té la certesa de que allò que s'ha implementat, funciona. Tanmateix, s'evita que les següents funcionalitats i recursos que utilitzen com a punt de partida necessari per a seguir escalant i creixent, són una base correcta i formen part d'una estructura sòlida.

També es troba el següent cas d'ús que ha estat útil. Hi ha moments en que s'executa una crida al servei i el resultat d'aquesta és negatiu. Gràcies a que s'han guardat les diferents crides que s'han fet i es té clara l'arquitectura del sistema, és fàcil depurar els problemes i veure que si l'error no prové directament de la crida que s'executa, prové d'altres crides implicades que s'han anat fent abans que aquella i es desfà el camí (amb les crides guardades) fins a trobar el punt conflictiu.

11 Conclusions

En aquest capítol final de la memòria del projecte, treure'm les conclusions a les que hem arribat sobre l'assoliment dels objectius, expectatives inicials, el producte obtingut i la valoració general.

11.1 Avaluació del potencial del producte generat

Considerem que aquest apartat és molt important i d'especial interès de cada a poder fer servir aquest producte, ja sigui per fer-ne ampliacions, per motius de manteniment o per integrar-la al sistema de monitorització de xarxes socials del qual ha sorgit la idea de realitzar aquest projecte. Si aquest TFG s'hagués presentat a principis de l'any 2018, seria completament diferent pel que respecta l'ús de l'API externa oficial d'Instagram, que no seria la Graph API. Degut a això, a que aquest món és molt canviant, considerem d'especial interès aquest apartat.

Davant del resultat derivat d'aquest TFG i les eines implementades, podem dir que s'ha complert amb l'arquitectura que es reflexa en el document. S'ha implementat el monitor amb èxit.

En resum, s'ha realitzat la feina relativa a tenir un monitor que funciona adequadament, ens hem adaptat davant els canvis que s'han fet per les APIs oficials de tercers i s'ha intentat trobar les millors respostes possibles davant les adversitats que han sorgit.

11.2 Avaluació general

Tenint en compte els apartats anteriors d'aquest punt del document, podem dir que s'han complert els objectius d'aquest projecte i que s'han aplicat satisfactòriament els coneixements adquirits durant els estudis de Grau en Enginyeria Informàtica. No obstant, com a la vida real, s'ha intentat donar solució als imprevistos que poden sorgir en tot projecte.

També s'han hagut d'aprendre tecnologies de les quals no es tenia el domini, l'ús de llibreries no conegudes, configuració de projectes a implementar i trobar solució a errors de configuració de software. Quan s'han tingut problemes d'eficiència, hem hagut de comprovar amb les eines de desenvolupador del navegador on hi havia el coll d'ampolla

Joan Colom Montoya – FIB UPC

i replantejar-nos coneixements adquirits fins al moment, amb l'objectiu de millorar l'eficiència. És a dir, el dia a dia d'un professional de la branca d'Enginyeria del Software.

12 Referències

- [1] DOMÍNGUEZ, Fernando; LÓPEZ, Rocío. Uso de las redes sociales digitales entre los jóvenes universitarios en México. Hacia la construcción de un estado del conocimiento (2004-2014). *Revista de comunicación*, 2015, no 14.
- [2] LIU, Bing. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 2012, 5.1: 1-167.
- [3] (GUADALUPE, Guadalupe Aguado; GARCÍA, Alberto García. Del Word-of-mouth al Marketing viral: aspectos claves de la comunicación a través de redes sociales. *Comunicación y hombre*, 2009, no 5, p. 41-51.
- [4] SALEHAN, Mohammad; NEGAHBAN, Arash. Social networking on smartphones: When mobile phones become addictive. *Computers in Human Behavior*, 2013, vol. 29, no 6, p. 2632-2639.
- [5] VELEV, Dimiter; ZLATEVA, Plamela. Use of social media in natural disaster management. *Intl. Proc. of Economic Development and Research*, 2012, 39: 41-45.
- [6] SINHA, Raj. Social network activity monitoring and automated reaction. U.S. Patent Application No 12/837,090, 2012.
- [7] NILSSON, Nils J.; NILSSON, Nils Johan. *Artificial intelligence: a new synthesis*. Morgan Kaufmann, 1998, p. 1.
- [8] RIDGWAY, Jessica L.; CLAYTON, Russell B. Instagram unfiltered: Exploring associations of body image satisfaction, Instagram# selfie posting, and negative romantic relationship outcomes. *Cyberpsychology, Behavior, and Social Networking*, 2016, 19.1: 2-7.
- [9] Grow your community with Hootsuite in <http://signuptoday.hootsuite.com/pro-esp-spanish-br-pplans/> (last accessed 23/10/17).
- [10] Social Listening Trusted by the World's Biggest Brands | Brandwatch in <https://www.brandwatch.com> (last accessed 23/10/17).
- [11] Media monitoring api – Mention in (<https://mention.com/es/media-monitoring-api/>) (last accessed 23/10/17).
- [12] OHNO, Taiichi. *Toyota production system: beyond large-scale production*. crc Press, 1988.

- [13] HERNÁNDEZ, Bali Isai Escalante, et al. Sustentabilidad: logística empresarial y manejo de logística inversa. *Observatorio de la Economía Latinoamericana*, 2013, no 185.
- [14] WANG, Xiaofeng; CONBOY, Kieran; CAWLEY, Oisin. “Leagile” software development: An experience report analysis of the application of lean approaches in agile software development. *Journal of Systems and Software*, 2012, vol. 85, no 6, p. 1287-1299.
- [15] SCHONBERGER, Richard J. Applications of single-card and dual-card kanban. *Interfaces*, 1983, vol. 13, no 4, p. 56-67.
- [16] FAVARE, J. Managing requirements for business value. *IEEE software*, 2002, vol. 19, no 2, p. 15-17.
- [17] Robert Walters salary survey 2016 in <https://www.robertwalters.co.uk/content/dam/salary-survey-2016.pdf> (last accessed 09/10/17).
- [18] STANDISH GROUP et al. “CHAOS manifesto 2013”. A: The Standish Group International. EUA (2011).