redditWho

Memòria del TFG (PAC Final)

Grau de Multimèdia
Àrea d'aplicacions interactives
2on semestre 2015-16

Autor: Ramon Royo i Giner
Consultor: Kenneth Capseta i Nieto
Professor: Carlos Casado i Martínez

20 de Juny de 2016

redditwho.com

LLICÈNCIES I PROGRAMARI

Llicències del codi de tercers utilitzat en el projecte

API de Reddit, llicència pròpia.

Bootstrap, llicència MIT.

Laravel, llicència MIT.

Timesearch.py, llicència MIT.

Guzzle, llicència MIT.

Programari utilitzat en el desenvolupament del projecte

Git, control de versions

Sublime Text 3, editor de text

Photoshop CS 6, editor gràfic

Microsoft Word 2013, processador de textos.

Microsoft Excel 2013, fulls de càlcul.

Filezilla, client de ftp.

Pàgina en blanc

DEDICATÒRIA

A la Marta, gràcies pel teu amor sense fi, la teva confiança en mi supera qualsevol barrera i em fa veure més enllà de les mancances imposades per la falta de confiança en mi mateix, en aquells moments d'ofuscació desmesurada que tu saps que tinc esporàdicament, quan els reptes em semblen impossibles.

Gràcies per ser-hi. T'estimo.

Al Pol, sempre et diem que el treball ben fet, la constància i l'esforç són necessaris per obtenir allò que desitgem, aquí en tens un petit exemple. Aquest projecte representa la suma de moltes hores d'estudi i la fi d'un procés que va començar fa més de 5 anys. Ets molt intel·ligent i et seguirem educant perquè obtinguis la millor versió de tu mateix. Encara que no ho apreciïs sempre, ho fem amb tot l'amor del món.

Nyu!

ABSTRACT

RedditWho és un servei desenvolupat per a la plataforma web *Reddit*, la seva finalitat és presentar una llista amb els usuaris i llocs web, que més vots positius hagin rebut en un interval de temps determinat.

RedditWho està dirigit principalment a aquells usuaris que creïn contingut per llocs web, com ara blocs i d'altres tipus de plataformes on el contingut sigui la base de generació de tràfic, i l'objectiu és posicionar el servei com a font de referència per poder trobar fàcilment publicacions —ja avaluades per milers d'usuaris a través del sistema de votacions de *Reddit*—, que serveixin d'inspiració per tal de crear nou contingut.

Els motius que m'han decidit dur a terme aquest projecte son diversos, el més important en l'apartat tècnic és la diversitat de tecnologies emprades en la creació de tot el servei i en l'aspecte personal, d'una banda l'oportunitat d'aprendre a utilitzar més profundament algunes de les eines emprades durant el grau i de l'altra, la motivació que em produeix crear un servei des de zero, publicar-lo a Internet i oferir-lo a milers d'usuaris potencials.

Paraules clau: *Reddit*, *Laravel*, Contingut, *PRAW*, *Python*, *Bootstrap, PHP, HTML5, CSS3*.

ABSTRACT (ENGLISH)

RedditWho is a service developed for *Reddit*, its purpose is to present a list of users and web sites that have topped their *subreddits* in a given time.

RedditWho is aimed primarily at users who create content for blogs and other platforms where the content is the basis of traffic generation, and the goal is to position the service as a source of reference for easily finding publications — evaluated by thousands of users through *Reddit's* voting system— that will serve as inspiration to create new content.

The reasons for choosing to develop this project are quite diverse, the most important from a technical point of view, is the diversity of technologies I'll be using to develop the service. On the personal side, for once the opportunity to learn more about some of the tools I've been using during the degree and on the other hand, it's a big motivation to create a service from start, publish it on the Internet and let thousands of potential users use it.

Keywords: *Reddit*, *Laravel*, Content, *PRAW*, *Python*, *Bootstrap, PHP, HTML5, CSS3*.

NOTACIONS I CONVENCIONS

# Títols nivell 1 dels temes	Nunito 22
## Títols nivell 2 dels temes	Nunito 18
Títols dels índexs	Nunito 22
Anglicismes i noms de marques	Verdana 11, cursiva
Adreces de pàgines web	Verdana 11
Text normal	Verdana 11
TÍTOLS DELS TEMES INTRODUCTORIS	Nunito 12
Capçalera	Verdana 9
Peu	Verdana 9, negreta
Peu d'imatges i taules	Verdana 8
Contingut taules	Verdana 9

```
user_agent = ("redditWho bot v0.0.1")
```

Prestige Elite Std 9

Taula de continguts

Índex de figures

Índex de taules

1. Introducció

Aquest projecte sorgeix d'un bloc anomenat *ViperChill* que fa temps que llegeixo. En aquest bloc el seu autor publica articles sobre el màrqueting a Internet.

Concretament aquesta idea prové d'una sèrie de 8 articles sobre idees de negoci, que segons l'autor, li hagués agradat dur a terme, de no ser perquè ja té tot el seu temps invertit en d'altres projectes. En el primer dels articles l'autor fa una anàlisi de les estratègies utilitzades per alguns usuaris de la plataforma *Reddit*, per tal de generar tràfic cap als seus llocs web, per mitjà de la publicació de contingut d'interès per als usuaris del *subreddit* —categories a *Reddit*, p.ex. */r/todayilearned* o */r/funny*—en que s'especialitzin.

A partir d'aquí, dóna algunes idees sobre com gestionar i aprofitar aquest tràfic, l'última d'aquestes l'enfoca de cara a donar un servei als usuaris que generen contingut, aprofitant les publicacions altament curades —per mitjà dels vots dels usuaris— de *Reddit*.

El servei, segons es planteja a l'article, consistiria en extreure de cada *subreddit* important, les 1000 publicacions de tots els temps amb més vots positius i crear una llista amb les adreces dels llocs web que les hagin generat.

Avaluant la possibilitat de realitzar aquest projecte, vaig visitar una pàgina anomenada *Reddit List*, que també aprofita l'API de *Reddit*. Aquesta pàgina en una sola plana mostra 3 columnes diferents, on es poden veure diferents *subreddits* i el nombre de subscriptors, la velocitat de creixement i d'altres dades estadístiques d'interès sobre la plataforma *Reddit*.

Tot i que a priori em va semblar un servei de relativament poc interès, vaig poder comprovar per mitjà de serveis que reporten el nombre de visites mensuals estimades, que el lloc rep entre 600.000 i 1.200.000 visitants mensuals. Aquestes xifres, junt amb l'oportunitat d'aprenentatge que m'aporta el projecte plantejat al bloc, em van fer decidir dur-lo a terme.

2. Definició del projecte

RedditWho s'emmarca dins de l'àrea de desenvolupament d'aplicacions interactives del Grau de Multimèdia de la UOC.

El projecte engloba tot el procés de creació d'un servei web, dissenyat per oferir a l'usuari informació destil·lada a partir d'una font d'informació, amb un gran volum de contingut.

Per tal de poder desenvolupar el servei, treballaré en profunditat una sèrie de coneixements adquirits durant el grau i d'altres novells.

Pel que fa a la part del servidor, faré ús del llenguatge de programació *Python* v3.5.1 —per a la comunicació amb *l'API* de *Reddit,* mitjançant un *wrapper* anomenat *PRAW v.3.4.0*—, i d'un *framework* de PHP anomenat *Laravel v.5.3*, per crear el servei web. La informació extreta de *Reddit* serà emmagatzemada en una base de dades relacional, per tal d'evitar consultes excessives i redundants a l'API de *Reddit*, el gestor de base de dades utilitzat serà *MariaDB v.10.1.9*.

Aquesta base de dades l'hauré de dissenyar de manera que s'emmagatzemi la informació d'una manera eficient i sense redundàncies.

I pel que fa a la part del client, utilitzaré els llenguatges *HTML5* i *CSS* i novament el *framework Laravel*, que fa ús del paradigma MVC i per tant, em serà útil a l'hora de dissenyar les vistes i la lògica del servei. A fi de facilitar-me la feina d'aplicar estils CSS a la web, utilitzaré el *framework Bootstrap* v3.3.6.

També tindré en compte els conceptes apresos sobre usabilitat i presentació de la informació, per tal que el servei sigui usable i fàcil d'entendre.

Finalment, tots els arxius del servei seran emmagatzemats en dos servidors virtuals privats (VPS), un acollirà el bot i la base de dades i l'altre la pàgina web sobre la que els visitants realitzaran la navegació. L'empresa triada per contractar els servidors s'anomena *Vultr Holdings LLC.*, i és accessible a través de la pàgina Vultr.com.

3. Objectius

3.1. Principals

Els objectius principals del TFG present són els següents:

- Desenvolupar un robot amb *Python*, per extreure la informació de la base de dades de *Reddit* detallada en els apartats anteriors, fent ús d'un *wrapper* anomenat *PRAW*, que facilita la comunicació amb *l'API* de *Reddit*.
- Dissenyar i implementar una base de dades per emmagatzemar la informació extreta de *Reddit*, per tal d'evitar consultes repetides, donat que tenen un límit de 60 consultes per minut a la seva *API*.
- Programar el servei que permeti extreure informació de la base de dades local i presentar-la a l'usuari de la manera adequada.
- Mostrar la informació a l'usuari de manera clara i fàcil d'entendre.
- Elaborar la memòria del projecte seguint la pauta prefixada.
- Elaborar la resta d'elements (vídeo, presentació visual/escrita).

3.2. Personals

Aquest serà el primer projecte seriós que desenvolupo i que veurà la llum pública, a més abasta una sèrie de coneixements tècnics en els que hauré d'aprofundir, per tal de poder completar els objectius exposats en els dos punts anteriors.

És per tant, un repte tècnic i organitzatiu que afronto amb ganes de dur a terme amb èxit, donats els beneficis personals i curriculars que em reportarà la seva consecució.

4. Escenari

La producció de contingut de qualitat o que resulti atractiu per al públic que el consumeix, és un factor determinant en l'èxit d'un lloc web. Aquest es traduirà en un nombre elevat de visites i això suposa un increment directe en els ingressos, per publicitat o venda d'articles, que pot generar el lloc web en qüestió.

La finalitat del meu projecte, consisteix en facilitar la identificació d'aquells llocs web i individus, que produeixen contingut que els visitants de *Reddit* valorin més positivament, dintre del context individual de cada fòrum. Així com identificar aquelles publicacions que més valoracions positives hagin rebut.

Amb aquest objectiu, es pretén facilitar a l'usuari visitant de la meva pàgina web, una sèrie de publicacions organitzades per interessos (*subreddits*) i vots, a banda d'una sèrie d'estadístiques, com ara el nombre d'usuaris subscrits a cada *subreddit*, les hores i dies en que les publicacions hagin rebuts més votacions independentment del seu contingut, quins usuaris i llocs web produeixen més contingut que es valora positivament, etcètera.

D'aquesta manera, es proporcionaran idees per produir contingut que consumeixen els usuaris interessats en una temàtica concreta. I així, es facilitarà la tasca creativa de generar nou contingut per blocs i pàgines que centrin el seu model de negoci, en la producció periòdica de contingut nou.

Per tal de facilitar la identificació d'idees noves i recents, es facilitaran una sèrie d'eines, com ara la possibilitat de llistar les publicacions emmagatzemades d'un únic *subreddit*, organitzar-les pel nombre de vots i per les dates de publicació, usuari que publica, lloc web, etcètera.

5. Continguts

5.1. Continguts generals

Els continguts del web consisteixen en una informació destil·lada, a partir d'una font d'informació molt més àmplia, com és *Reddit*.

Concretament es mostren els 1500 *subreddits* amb més subscriptors, junt amb dades com ara la data de creació del grup, descripció i la seva adreça, entre d'altres. D'aquests a banda, s'extreuen dades sobre les publicacions amb més vots i es mostren a l'usuari.

Degut a les limitacions inherents a *l'API* de *Reddit*, que limita el temps entre peticions al seu contingut a 1 segon, l'extracció de tot el contingut es realitza de forma periòdica i de manera asíncrona al servei web, per mitjà de scripts programats amb *Python*, que guarden tota la informació actualitzada en una base de dades, des d'on el servei extreu el contingut i el presenta als usuaris.

5.2. Estructura

Inici

Es carrega una llista amb els 25 *subreddits* amb més subscriptors, junt amb dades sobre aquests. La vista està paginada de manera que es pot navegar endavant i endarrere, per tal de mostrar la resta de *subreddits*. Els noms són clicables i porten a vistes individuals de cadascun.

Subreddits

En aquesta vista, de cada *subreddit* es carreguen les publicacions amb més vots. S'hi inclou el nom de l'usuari que ha fet la publicació i si el contingut no és un *selfpost* (publicació a *Reddit*), també s'inclou l'adreça del lloc web enllaçat.

Els noms dels usuaris són clicables i enllacen al seu perfil de *Reddit*. Els noms dels llocs web enllacen amb la seva pàgina i els títols de les publicacions enllacen amb la pàgina corresponent a *Reddit*, o bé amb la publicació externa a *Reddit*, en cas de que no es tracti d'una publicació de text hostatjada a *Reddit*.

6. Metodologia

La metodologia emprada per realitzar el projecte és la del desenvolupament i consecució de les tasques de manera seqüencial, i paral·lela en alguns casos, segons s'ha planificat al diagrama de *Gantt*.

Per tal de completar el projecte, a banda del treball sobre la tasca en si mateixa, s'intercala la recerca sobre els diferents aspectes específics que composen cadascuna, com ara el control de versions per mitjà del programari *Git*, el disseny correcte d'una *BBDD* relacional, el disseny d'un robot amb *Python* per comunicar-se amb *l'API* de *Reddit*, i d'altres tasques, que no formen part dels coneixements específicament treballats durant el grau.

A cada entrega es revisa aquesta memòria amb la finalitat d'actualitzar-ne el contingut, amb noves aportacions o canvis necessaris, esdevinguts a causa de la naturalesa del projecte.

7. Arquitectura del servei

7.1. Client

L'usuari final realitzarà la consulta de la informació oferta pel servei, a través del seu navegador web, per tant no hi ha cap requeriment especial, a banda d'un navegador web actualitzat capaç de mostrar contingut en HTML5 i CSS3.

7.2. Servidors

Els servidors que gestionaran el contingut del servei, són servidors virtuals privats o VPS, amb el sistema operatiu *CentOS 7 x64* instal·lat i el servidor web *Apache v2.4.6*, així com amb suport per *PHP v7.0*, *Python v3.5.1* i SQL amb el gestor *MariaDB v10.1.9*.

Els recursos dels VPS consisteixen en un processador d'un nucli, 768MB de memòria RAM i 1TB de dades mensuals. Aquests recursos són ampliables en qualsevol moment.

7.3. Base de dades

La base de dades està formada per cinc taules. A continuació es detallen de forma general les seves funcions:

- ***subreddits***, emmagatzema informació sobre els *subreddits*.
- ***posts***, emmagatzema informació sobre les publicacions amb més de 500 vots.
- ***subscribers***, emmagatzema informació sobre l'evolució temporal del nombre de subscriptors, dels *subreddits* més poblats.
- ***excepts****, conté missatges generats per excepcions produïdes durant l'execució del codi del bot.*
- ***latestsposts****,* s'hi guarden de cada *subreddit*, les últimes dates consultades pel bot durant la darrera exploració. Així és possible començar la propera exploració, des del moment en que va acabar la última.

La definició de les taules utilitza el motor *InnoDB*, i el joc de caràcters *utf8mb4* amb la intercalació (*collate*) *utf8mb4_unicode_ci*.

A part de les claus primàries de cada taula, he creat uns índexs específicament pensats per facilitar la recuperació de dades de les taules, a continuació es mostren les diferents columnes amb els tipus associats, així com els índexs.

Field	Type	Null	Key	Default	Extra
idstr	char(6)	NO	PRI	NULL	
idint	int(10) unsigned	NO	UNI	NULL	
display_name	varchar(30)	NO		NULL	
created_utc	int(10) unsigned	NO		NULL	
description	varchar(500)	NO		NULL	
subscribers	int(10) unsigned	NO		NULL	
submissions	int(10) unsigned	NO		0	
over18	tinyint(1)	NO		NULL	
updated	timestamp	NO		CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP

Taula 7.1. MySQL taula subreddits

Field	Type	Null	Key	Default	Extra
idstr	char(6)	NO	PRI	NULL	
idint	int(10) unsigned	NO	MUL	NULL	
title	varchar(300)	NO		NULL	
author	varchar(30)	NO	MUL	NULL	
subreddit	varchar(30)	NO		NULL	
score	smallint(5) unsigned	NO		NULL	
ups	smallint(5) unsigned	NO		NULL	
downs	smallint(5) unsigned	NO		NULL	
num_comments	smallint(5) unsigned	NO		NULL	
is_self	tinyint(1)	NO		NULL	

domain	varchar(191)	YES		NULL	
url	varchar(5000)	YES		NULL	
created_utc	int(10) unsigned	NO		NULL	
over18	tinyint(1)	NO		NULL	
updated	timestamp	NO		CURRENT_TIMESTAMP	onupdate CURRENT_TIMESTAMP

Taula 7.2. MySQL taula posts

Field	Type	Null	Key	Default	Extra
id	int(10) unsigned	NO	PRI		auto_increment
idsub	int(10) unsigned	NO	MUL		
subscribers	int(10) unsigned	NO			
updated	timestamp	NO		CURRENT_TIMESTAMP	onupdate CURRENT_TIMESTAMP

Taula 7.3. MySQL taula subscribers

Field	Type	Null	Key	Default	Extra
id	int(10) unsigned	NO	PRI		auto_increment
description	varchar (2000)	NO	NULL		
updated	timestamp	NO		CURRENT_TIMESTAMP	onupdate CURRENT_TIMESTAMP

Taula 7.4. MySQL taula excepts

Field	Type	Null	Key	Default	Extra
idsub	int(10) unsigned	NO	PRI		
lastDateUTC	int(10) unsigned	NO			

Taula 7.5. MySQL taula latestposts

8. Plataforma de desenvolupament

8.1. Programari

Cal diferenciar entre el programari utilitzat localment i el programari utilitzat en els servidors on s'allotjaran la base de dades, el bot i la pàgina web.

Local

Windows 10 Edu	Sistema operatiu.
mRemoteNG	Gestor de connexions remotes, utilitzat per connectar via protocol SSH amb els servidors remots i poder executar ordres de consola.
Filezilla	Client FTP utilitzat per pujar els arxius locals als servidors.
Sublime Text 3	Editor de textos amb coloració automàtica del text, que facilita la lectura del codi font.
Python 3.5.1	Intèrpret del codi font *Python* utilitzat en el desenvolupament del bot que explora els *subreddits*.
MariaDB 10.1.9	Gestor de bases de dades utilitzat per emmagatzemar les dades extretes de *Reddit*.
PHP 7	Intèrpret del codi font PHP utilitzat en el desenvolupament de la part del servidor de la pàgina web.
Laravel 5.1	*Framework* de PHP utilitzat en el desenvolupament de la part del servidor de la pàgina web.
Chrome	Navegador web utilitzat per comprovar el codi de la pàgina web i realitzar les cerques per documentar-me.

Postman	Aplicació de Chrome utilitzada per comprovar la sortida en format JSON de l'API REST.

Taula 8.1.1. Programari local

Servidors, programari comú

CentOS 7 x64	Distribució del sistema operatiu Linux.
Apache 2.5.4	Servidor web.
PHP 7	Intèrpret del codi font PHP utilitzat en la generació i la consulta de les peticions a l'API REST.
Laravel 5.1	*Framework* de PHP utilitzat en el desenvolupament de la part del servidor de la pàgina web.
Servidor ftp	Per poder connectar i pujar els arxius locals.
Firewalld	Per protegir els servidors d'atacs externs.

Taula 8.1.2. Programari comú dels servidors

Servidor base de dades

MariaDB 10.1.9	Gestor de bases de dades utilitzat per emmagatzemar les dades extretes de Reddit i permetre'n la consulta.
Python 3.5.1	Intèrpret del codi font *Python* utilitzat en el desenvolupament del bot que explora els *subreddits*.

Taula 8.1.3. Programari específic del servidor de la base de dades

Servidor pàgina web

En aquest servidor s'allotjarà únicament la pàgina web que veuran els visitants del lloc web, aquesta realitzarà peticions a l'API REST que estarà funcionant al servidor de la base de dades i a continuació, mostrarà aquestes peticions formatades a l'usuari.

8.2. Maquinari

Novament, cal diferenciar entre el maquinari local i el maquinari dels servidors on s'allotjaran la base de dades, el bot i la pàgina web.

Local

Processador	Pentium Dual-Core E6600 3.06GHz
RAM	4GB
Disc dur 1	Samsung SSD 850 EVO 250GB
Disc dur 2	Samsung HD103SJ ATA 1TB
Tarja gràfica	AMD Radeon HD 5670

Taula 8.2.1. Maquinari local

Servidors

Inicialment els dos servidors remots utilitzaran la mateixa configuració de maquinari, un *VPS* contractat a l'empresa *Vultr Holdings LLC*, amb un cost mensual de 5€ cadascun. A mesura que es detecti una necessitat més elevada de maquinari, s'anirà ampliant la contractació de configuracions més potents.

Processador	Intel, processador virtual a 2.4GHz.
RAM	768MB
Disc dur	15GB
Dades mensuals	1000TB

Taula 8.2.2. Maquinari servidors

9. Planificació

9.1. Dates Clau

Les dates clau es corresponen amb les dates de les PACs que cal anar entregant durant el semestre, es troben detallades a continuació:

PAC	Data
PAC1	08/03/2016
PAC2	06/04/2016
PAC3	08/05/2016
Lliurament final	20/06/2016

Taula 9.1. Dates clau

9.2. Fites

Les etapes generals en que es subdivideix el projecte, es poden concretar en les següents activitats:

- Definició del projecte.
- Disseny de la base de dades.
- Programació del robot que extraurà la informació de la base de dades de *Reddit* i l'emmagatzemarà a la BBDD local.
- Disseny i desenvolupament del servei que interactuarà amb la base de dades i mostrarà la informació a l'usuari.
- Disseny i desenvolupament del servei encarregat de la visualització de dades.
- Tests del servei, per trobar i eliminar errors i avaluar la usabilitat.

9.3. Diagrama de Gantt

#	Tasca	Inici	Final	Dep.	Dies
1	**TFG RedditWho**	**02/24/2016**	**06/20/2016**		**117**
1.1	**PAC1**	**02/24/2016**	**03/08/2016**		**13**
1.1.1	Definició del projecte	02/24/2016	03/04/2016		9
1.1.2	Pla de treball	03/04/2016	03/08/2016		4
1.1.3	Redacció de la memòria	02/24/2016	03/08/2016		13
1.2	**PAC2**	**03/08/2016**	**04/06/2016**	**1.1**	**29**
1.2.1	Actualització memòria	03/08/2016	04/06/2016		29
1.2.2	Tria del servidor i domini	03/08/2016	03/09/2016		1
1.2.3	Disseny i creació *BBDD* (*MySQL*)	03/08/2016	03/12/2016		4
1.2.4	Robot *API reddit* (*Python*)	03/12/2016	04/06/2016	1.2.3	25
1.3	**PAC3**	**04/06/2016**	**05/08/2016**	**1.2**	**32**
1.3.1	Actualització memòria	04/06/2016	05/08/2016		32
1.3.2	Aprenentatge Laravel i API REST	04/06/2016	04/22/2016		16
1.3.3	Configuració servidor BBDD i instal·lació del bot.	04/18/2016	04/20/2016		2
1.3.4	Seguiment del funcionament del bot i correccions.	04/20/2016	04/22/2016		2
1.3.5	Programació API REST (*Laravel*)	04/23/2016	05/08/2016	1.3.2	12
1.4	**Entrega Final**	**05/08/2016**	**06/20/2016**	**1.3**	**43**
1.4.1	Actualització memòria	05/08/2016	06/20/2016		43
1.4.2	Programació API REST (*Laravel*)	05/08/2016	05/15/2016		7
1.4.3	Programació Web (*Laravel*)	05/15/2016	05/29/2016	1.4.2	14
1.4.4	Tests	05/29/2016	05/30/2016	1.4.3	1
1.4.5	Presentació públic general	05/30/2016	06/14/2016	1.4.4	14
1.4.6	Presentació vídeo tribunal	05/30/2016	06/14/2016	1.4.4	14
1.4.7	Autoinforme	06/14/2016	06/18/2016	1.4.4	4
1.4.8	Entrega i publicació *O2*	06/18/2016	06/20/2016	1.4.7	2

Taula 9.2. Tasques (Gantt)

10. Procés de desenvolupament

10.1. Domini i allotjament

El domini s'ha contractat amb el registrador de dominis Namecheap i l'adreça és www.redditwho.com.

L'allotjament està proporcionat per l'empresa Vultr i s'ha contractat el servei mínim que inclou un VPS amb accés a una CPU, 768MB de RAM, 15 GB d'emmagatzematge en un disc dur SSD i 1 TB mensual de transferència de dades. El servidor no està gestionat per l'empresa, s'ha configurat des de zero amb el sistema operatiu *CentOS 7 x64* i el programari necessari per poder servir pàgines web dinàmiques.

10.2. Base de dades

El següent pas ha estat dissenyar una base de dades que permeti emmagatzemar la informació capturada. El disseny inicial s'ha anat redefinint a mesura que s'han reavaluat les necessitats del projecte.

A la pràctica, un cop s'han extret les primeres mostres de la web de *Reddit*, s'han fet canvis a les taules que emmagatzemen aquesta informació, concretament als tipus de les dades i s'ha afegit columnes, que inicialment no s'havien considerat.

També s'han afegit índexs, inicialment no contemplats. Donada la necessitat de realitzar consultes amb diferents valors d'ordenació.

10.3. Captura del contingut de Reddit

La captura del contingut de *Reddit*, s'ha desenvolupat en tres fases diferenciades.

A la primera s'han desenvolupat en *Python*, els scripts necessaris per dur a terme l'autorització de l'aplicació perquè pugui fer ús del compte d'usuari de *Reddit*, per crear els *tokens* d'identificació a l'*API* i per poder connectar amb la base de dades i amb *Reddit*, fent ús d'una sola línia de codi.

La segona fase ha consistit en el desenvolupament del script encarregat de capturar els més de 700.000 *subreddits* existents i emmagatzemar a la base de dades el nom, el codi d'identificació, la descripció, etc.

A la tercera i última fase, s'ha desenvolupat el script que captura les publicacions de cadascun dels *subreddits* seleccionats. A més, aquest script s'encarrega de cridar periòdicament al script desenvolupat a la segona fase, per tal de mantenir la llista de *subreddits* actualitzada.

10.4. Configuració dels servidors remots

Un cop s'ha instal·lat el sistema operatiu als VPS, cal realitzar un procés de configuració per poder-los utilitzar com a servidors web i de base de dades.

Amb la finalitat d'automatitzar aquest procés, he dedicat un temps a fer recerca i a escriure un script amb totes les ordres necessàries, per tal d'instal·lar tot el programari als servidors remots i realitzar canvis en la configuració, que incrementin la seguretat davant de possibles atacs externs.

10.5. Desenvolupament de l'API REST

Amb la finalitat de separar la base de dades de la pròpia pàgina web, per tal de facilitar l'escalat futur en cas de que sigui necessari i millorar la seguretat, he decidit desenvolupar una *API REST* amb *Laravel*, al servidor de la base de dades. Els retorns de les consultes realitzades a aquesta *API*, es troben formatats en JSON.

A aquesta *API* sols hi tindrà accés l'aplicació del servidor web i hi realitzarà consultes, el retorn de les quals formatarà adequadament i el presentarà a l'usuari de la pàgina web.

10.6. Desenvolupament de l'aplicació web

L'última fase del projecte correspon al desenvolupament de l'aplicació web, que realitzarà consultes a l'API i presentarà les dades dels *subreddits* i les publicacions als usuaris.

11. APIs utilitzades

API de Reddit

Per tal de poder capturar la informació de la pàgina de *Reddit*, cal fer ús de la seva API, dissenyada per facilitar l'extracció de contingut de la seva pàgina.

L'accés a aquesta *API* el realitzo per mitjà d'un *wrapper* anomenat *PRAW* i desenvolupat en *Python*. Aquest incorpora una sèrie de funcions que faciliten l'accés a l'*API*, així com també una sèrie de regles que implementen les normes d'accés, com ara limitar el temps entre consultes a 1 segon.

12. Perfils d'usuari

Els possibles usuaris del servei són els següents:

Propietari o contribuïdor a blocs o llocs web	Aquest usuari crea contingut per un lloc web o bloc i necessita idees per crear aquest contingut. El servei li permet tenir accés a una gran quantitat de material, organitzat per temàtica i filtrat pels vots positius dels usuaris de *Reddit*.
Usuari previ de Reddit	Aquest usuari és habitual de *Reddit* i visita el meu servei perquè té curiositat per veure estadístiques sobre els *subreddits*, com ara el nombre de publicacions totals, el nombre de publicacions amb un nombre de vots per sobre d'un valor concret, hores de publicació, quins usuaris i llocs web són els més votats, etcètera.
Usuari casual	Aquest usuari ha sentit a parlar del servei en algun fòrum o bloc i el visita per curiositat, de manera puntual.

Taula 12.1. Perfils d'usuari

13. Seguretat

La seguretat s'ha treballat des de dos enfocaments molt diferents, d'una banda la integritat de les dades emmagatzemades a la base de dades i de l'altra, la seguretat davant d'atacs de tercers.

13.1. Integritat de les dades emmagatzemades a la BBDD

Tot el codi que he desenvolupat en *Python*, inclou blocs *Try:Except* que permeten executar un bloc de codi i en cas de que es produeixi un error, es realitzin accions concretes per minimitzar els efectes d'aquest error. Les parts més delicades del meu codi, es corresponen a les consultes a *l'API* de *Reddit* i les escriptures a la base de dades.

En el primer cas, la llibreria *PRAW* ja inclou mecanismes propis, per repetir les consultes a l'API de *Reddit*, en cas de que les respostes no siguin les esperades.

En el segon cas, he programat el codi perquè les excepcions s'enregistrin a la taula *excepts* de la BBDD i així pugui fer un seguiment dels possibles errors que es produeixin durant l'execució del programa. En cas de que no es puguin fer escriptures o consultes a la BBDD, he programat el bot perquè es tanqui i així s'eviti fer exploracions a *Reddit*, que no quedin enregistrades a les corresponents taules de la BBDD.

Un cop el bot ha finalitzat una exploració completa dels *subreddits* d'interès, que li ha portat aproximadament 34 dies en total, he realitzat una còpia de seguretat de tota la base de dades, per tal de poder restaurar-la en cas de necessitat.

13.2. Seguretat dels servidors davant d'atacs de tercers

Per començar, actualitzo periòdicament el programari dels servidors, per tenir-los al dia i amb els errors corregits. A més, el sistema operatiu incorpora un tallafocs. Les claus d'accés són alfanumèriques i de 12 caràcters com a mínim.

També he separat la BBDD del servidor web que servirà el contingut als visitants, per evitar que tots dos llocs puguin quedar compromesos al mateix temps.

D'altra banda, he configurat el servidor web on és l'*API* de manera que tan sols el servidor de la pàgina web pugui fer-hi consultes.

Pel que fa al tipus d'accés a la interfície de línia d'ordres, sols és possible a través del protocol *SSH2*, que és semblant a la connexió *Telnet*, però està encriptada. La versió 1 d'aquest protocol, així com l'accés per *Telnet*, es troben desactivats per raons de seguretat.

A més, he eliminat l'accés SSH de l'usuari *root*, de manera que és impossible connectar als servidors amb aquest usuari, utilitzant l'accés remot.

Per últim, l'accés a la BBDD es realitza per mitjà de l'usuari *rwhobot*, que tan sols té accés a la base de dades *redditwho*.

14. Tests

14.1. Bot Python

He executat localment el codi múltiples vegades i he anat corregint tots els problemes detectats, quan he cregut que el programa estava preparat per a la fase de producció, l'he pujat al servidor de la BBDD i l'he deixat executant-se.

Després d'una primera iteració que ha durat 34 dies, no he detectat problemes greus en el funcionament del bot.

14.2. Servidor API

Seguint el mateix procés emprat amb el bot, he anat provant el codi fins a obtenir els resultats desitjats.

Actualment l'API funciona correctament i serveix el contingut tal com s'espera.

14.3. Servidor Web

He navegat per totes les pàgines i he fet ús de tots els enllaços disponibles, amb la finalitat de detectar errors i mancances.

15. Versions del servei

Alpha

- Disseny de la base de dades
- Desenvolupament del bot

Beta

- Proves del funcionament correcte del bot
- Població de la base de dades en producció
- Desenvolupament de l'API REST amb *Laravel*

Versió 1.0

- Desenvolupament de la pàgina web amb *Laravel*

16. Requisits

16.1. Client

Un ordinador, tauleta o mòbil capaços d'executar un navegador actualitzat i amb connexió a Internet. No és necessària cap instal·lació, ni cap coneixement específic per fer ús del servei web.

16.2. Servidors

En fase de desenvolupament, n'hi ha prou amb un sol processador de 2.4Ghz i 768MB de RAM. Tant el gestor de la BBDD, com el bot en *Python* funcionen correctament.

En fase de producció, serà necessari com a mínim duplicar la RAM del servidor de la base de dades i més enllà no ho puc prevenir, tot dependrà del nombre de visites mensuals que hagi de suportar el lloc web.

17. Bugs

17.1. Script de Python del bot

El primer i més important ha estat provocat pels caràcters no ASCII, que es troben en alguns dels títols de les publicacions de *Reddit*.

El puc dividir en dues fases, la primera es correspon a l'ús de cometes simples i dobles en els títols, que provoquen que s'enviïn consultes SQL errònies a la BBDD. Al trobar-se, el text de les consultes, envoltat per cometes simples, si el títol també les inclou, llavors es genera un error de *MySQL*.

La solució ha passat per crear una funció que escapi les cometes simples als títols, quan les trobi.

La segona fase, una evolució d'aquesta primera, l'han provocat altres caràcters *unicode* i les barres que s'utilitzen per escapar "\\" caràcters.

Després de moltes cerques, gràcies a una publicació de *Stackoverflow*, he descobert la llibreria de *Python* per treballar amb expressions regulars, anomenada *re*. Aquesta conté una funció, *escape()*, que aplicada a una cadena de text, escapa tots els caràcters no alfanumèrics que podrien provocar problemes com els que he tingut.

Gràcies a aquesta llibreria, he pogut substituir la funció que he creat a la fase 1 i eliminar per complet tots els problemes amb els títols.

A banda, altres *bugs* que he detectat, però que encara no he corregit, ja que no considero que tinguin importància immediata, són:

- El bot deixa passar 5 dies entre cada exploració, i d'aquests sols explora els 3 primers, deixant un marge de 2 dies entre la data actual i la data de finalització, amb la finalitat de donar temps a que les publicacions acumulin vots. Doncs bé, aquest marge de 2 dies no és respecta i és important que ho sigui, ja que sinó no s'emmagatzemaran publicacions que amb el marge correcte tindrien més de 500 vots, que és el límit que he establert.
- La descripció d'alguns dels *subreddits* no s'ha emmagatzemat a la base de dades. Hauré de comprovar individualment amb un script de *Python*

aquests *subreddits*, per veure si és un problema del meu bot o de l'API de *Reddit*, ja que visitant els grups he pogut apreciar que si que tenen una descripció.

17.2. Base de dades

El que al principi em va semblar un *bug* en una de la llibreries que utilitzo al bot, ha resultat ser un *bug* en el disseny de la BBDD.

El *bug* consisteix en que una funció retorna tipus diferents quan la consulta realitzada no troba cap coincidència, segons si aquesta s'executa sobre una taula o una altra. Bàsicament no té cap raonament lògic des del meu punt de vista, ja que la diferència en el tipus retornat no té cap justificació aparent, si em fixo solament en el codi *Python*.

El problema radica en la definició de les columnes consultades a la BBDD. Les definicions inicials eren les següents:

```
CREATE TABLE IF NOT EXISTS subreddits (
...
submissions   int unsigned NULL
...
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;

CREATE TABLE IF NOT EXISTS latestposts (
...
lastDateUTC   int unsigned NOT NULL DEFAULT 0
...
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

La solució ha consistit en canviar la definició de la columna *submissions* per fer-la igual que la definició de la columna *lastDateUTC*.

17.3. API i Web

Tots els *bugs* detectats han estat corregits.

18. Projecció a futur

La llista de modificacions futures és àmplia i se n'aniran afegint de noves, a mesura que el servei sigui utilitzat per usuaris. Moltes d'elles vindran sens dubte, suggerides pels propis usuaris.

A continuació llisto algunes de les que se m'han anat ocorrent durant el desenvolupament del projecte, i que no he implementat per falta de temps i/o coneixements.

Realitzar les càrregues de les llistes de *subreddits* i publicacions per mitjà d'*Ajax*. Ara mateix amb cada canvi de pàgina, tot el contingut s'ha de recarregar, això no és el més òptim per a l'experiència d'usuari.

Permetre la reordenació dels *subreddits* i de les publicacions per criteris diferents als definits per defecte. Aquesta modificació ja està implementada a l'API, on les rutes de consulta definides permeten enviar diferents paràmetres per modificar les consultes a la base de dades. Perquè els usuaris en puguin fer ús, falta implementar-ho a la Web.

Implementar el visionat dels resultats emmagatzemats a la taula *subscribers*, que permetrà observar l'evolució temporal del nombre de subscriptors en els *subreddits*.

Afegir un cercador de *subreddits* i autors. Per tal de poder-los trobar directament.

Implementar unes taules de rànquings on es mostrin llistes ordenades d'usuaris i dominis, segons diferents criteris de valoració, com ara el nombre total de publicacions, valor mitjà de puntuació, etc.

19. Anàlisi de mercat

El servei com a tal, no té una competència directa de cap altre lloc web, però si que existeixen d'altres serveis que basen el seu funcionament en el contingut de *Reddit*. El fet de la seva existència i la quantitat de visites mensuals, és una de les raons que em van fer decidir desenvolupar aquest projecte. Alguns d'ells els llisto a continuació.

RedditList

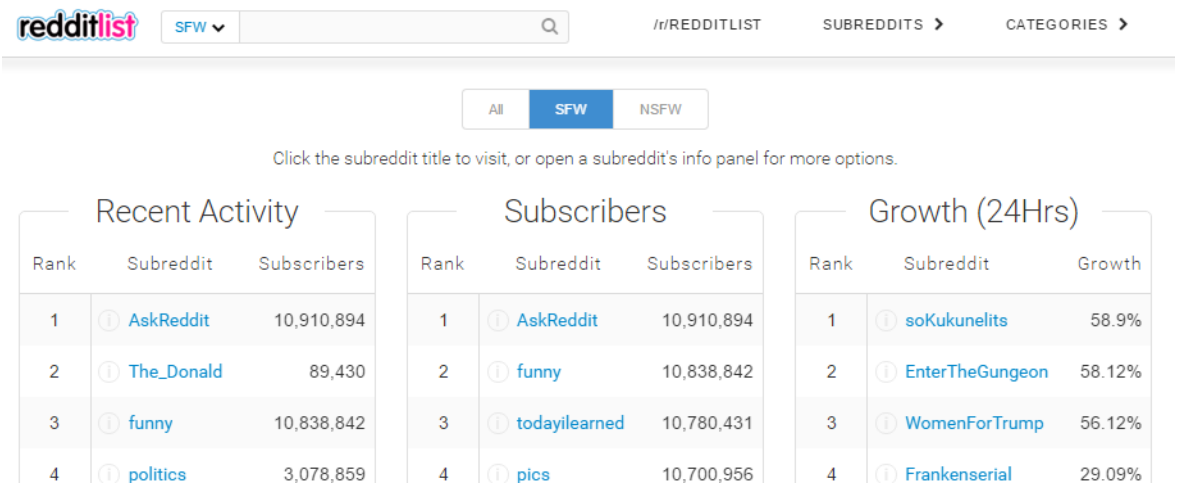

Figura 19.1. RedditList

Aquesta pàgina presenta tres taules i cadascuna mostra un tipus d'informació. En una es pot veure quins *subreddits* han tingut més activitat recentment, a una altra es veuen els *subreddits* ordenats pel nombre de subscriptors i a la tercera, es poden veure els *subreddits* amb un creixent més accelerat durant les últimes 24 hores.

Llevat d'això, la pàgina no ofereix massa més, però si posem la seva adreça a la pàgina SimilarWeb, per veure el seu posicionament mundial i el nombre de visites, s'aprecia que té un tràfic força important, rondant el milió de visites mensuals.

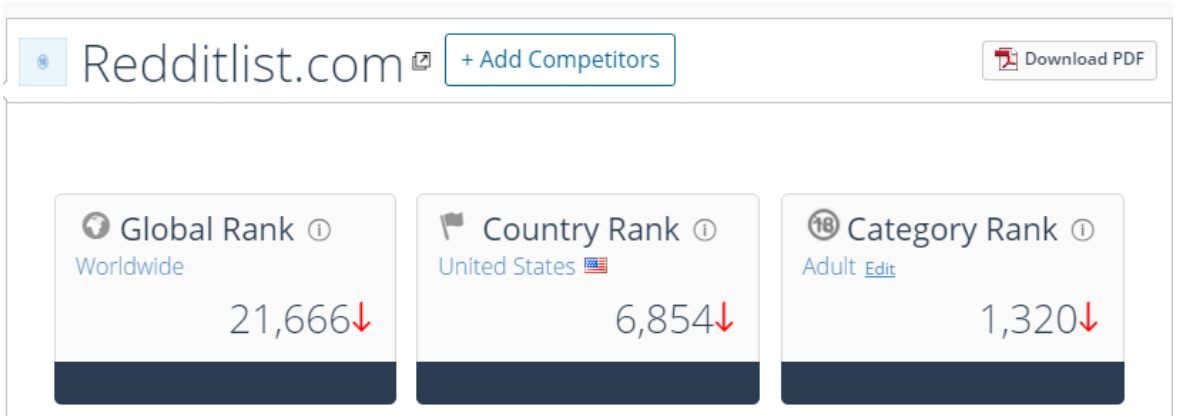

Figura 19.2. SimilarWeb -

RedditStatus

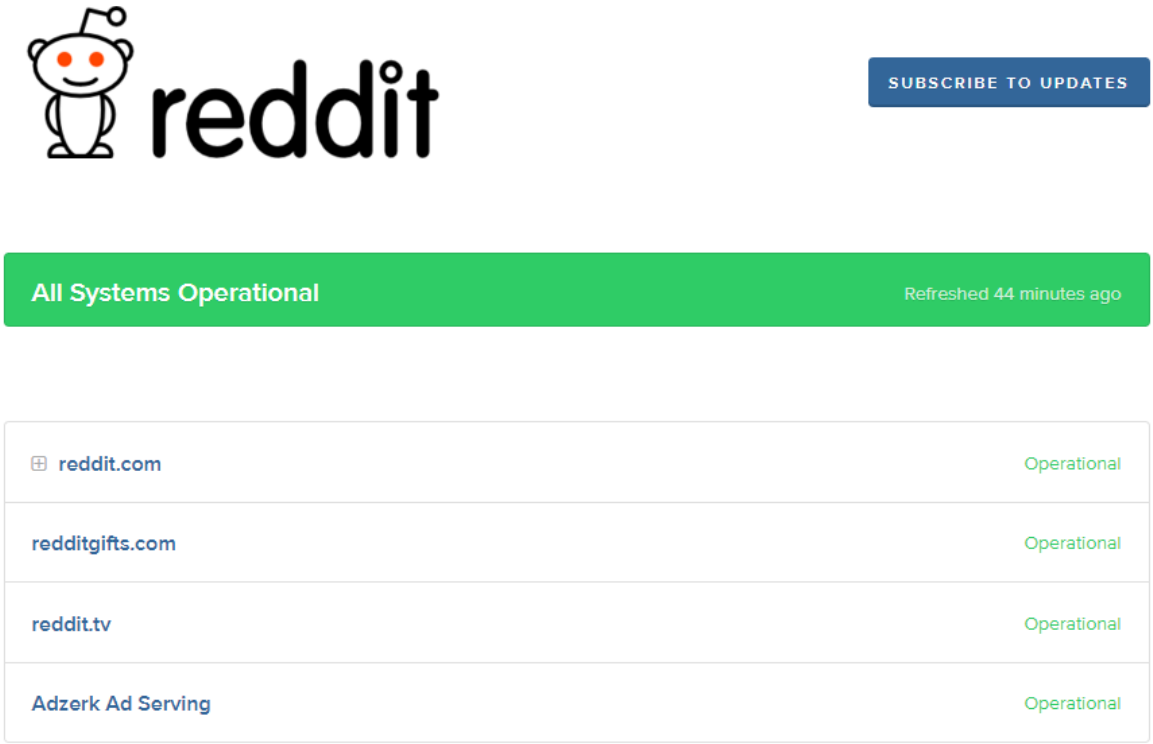

Figura 19.3. RedditStatus

Aquesta pàgina realitza un seguiment de l'estat de vàries pàgines relacionades amb *Reddit*, mostra gràfiques amb informacions com ara el nombre de peticions servides, el temps en línia, etc. Una visita a *SimilarWeb* ens revela que la pàgina rep unes 160.000 visites mensuals.

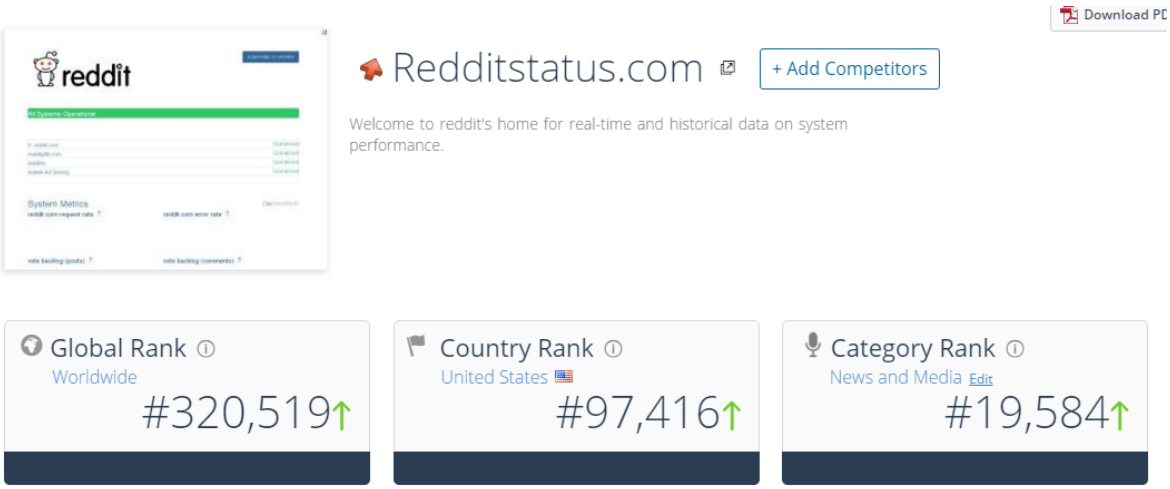

Figura 19.4. SimilarWeb - RedditStatus

Hi ha d'altres pàgines que ofereixen serveis al voltant de *Reddit*, no totes tenen els resultats que té *RedditList*, però demostren que hi ha un interès per aquest tipus de producte i fan que el projecte tingui un sentit, més enllà de la simple tasca de crear-lo.

20. Conclusions

El desenvolupament d'aquest projecte m'ha permès treballar amb un ventall de tecnologies molt divers, tant en l'àmbit del *back-end*, com en el del *front-end*.

He hagut d'orientar els meus esforços, de cara a obtenir un nivell de coneixement suficientment elevat de cadascuna d'aquestes tecnologies, per tal de poder dur a la totalitat del projecte. Tanmateix, he fet un ús extensiu de les cerques a Internet i a fòrums especialitzats en programació, com *Stackoverflow*.

Hi han hagut moments de frustració molt intensos, generats al enfrontar-me a reptes tècnics per als quals no tenia els coneixements suficients, com per exemple la correcta configuració dels servidors, per poder-hi executar els projectes de *Laravel*. Així com moments d'immensa alegria, per exemple i seguint amb el cas exposat, al veure el logotip de *Laravel* per primer cop a la finestra del navegador, al executar un codi des del servidor remot, després de 20h de proves i errors.

Estic molt satisfet amb els resultats obtinguts en les fases de desenvolupament del bot i de l'API, tot i que considero millorable la fase del desenvolupament de la Web, ja que hi ha moltes millores que per falta de coneixements del llenguatge PHP i del *framework Laravel*, no he estat capaç d'implementar.

En aquest projecte he treballat per primer cop, en el desenvolupament d'un producte produït per al consum massiu a Internet. Fent ús d'una sèrie de tecnologies, que permeten desenvolupar productes de qualitat i sobretot, d'utilitat per a l'usuari final, fet que he trobat en falta durant tot el grau. Això m'ha despertat un interès molt gran, per seguir formant-me en aquest àmbit de la programació i en el desenvolupament de serveis.

Com a conclusió final, vull destacar la importància que ha tingut per mi desenvolupar un producte sencer des de zero, ja que m'ha permès treballar competències transversals tant importants com la planificació i l'organització de la feina, la resolució de problemes i la capacitat per a aprendre noves disciplines.

Tot això m'ha permès identificar fortaleses i debilitats i ha reforçat la confiança en les meves capacitats, a més d'ampliar-les.

Annex 1. Lliurables del projecte

A1.1. Scripts del bot en Python

Aquests arxius es poden trobar a la carpeta "Python - RedditWho Bot".

oauth_token.py	Script d'un sol ús, sols cal executar-lo una única vegada, per crear un *token* de refresc i autoritzar l'accés del bot al compte de *Reddit*. El *token* de refresc permet crear nous *tokens* d'accés a l'API de *Reddit*, aquests últims tenen una vida de tan sols 60 minuts i cal regenerar-los. Ús, s'executa directament des de la interfície de línia d'ordres de *Ms-Dos* o *Linux*, obre una instància de navegador i després des de la interfície de línia d'ordres demana la introducció d'un codi. Llegir els comentaris en línia per més detalls.
loginData.py	Script on es guarden les dades d'accés a l'API de *Reddit* i la base de dades.
redditWhoBot.py	El bot que extreu les dades de *Reddit* i pobla les taules de la base de dades. Ús, des de la interfície de línia d'ordres, executar: >>> python redditWhoBot.py
utils.py	Script amb diferents funcions útils i d'ús comú.

Taula A1.1.1. Scripts del bot en Python

A1.2. Script de la base de dades

Aquests arxius es poden trobar a la carpeta "Base de Dades".

redditwho.sql	Definició de la base de dades i de l'usuari que en tindrà accés.
Backup_BBDD.sql	Còpia de la base de dades.

Taula A1.2.1. Scripts de la base de dades

A1.3. Script de configuració del servidor

Aquests arxius es poden trobar a la carpeta "Scripts VPS".

CentOS 7 x64 - configuració dels servidors.sh	Aquest script configura els servidors perquè pugui ser utilitzats com a servidor web i/o de base de dades.
els_meus_alias.sh	Un script amb àlies definits per facilitar la interacció amb els servidors.
startbot.sh	Un script per facilitar l'arrencada del script del bot.

Taula A1.3.1. Scripts de configuració del servidor

A1.4. Projectes Laravel de l'API REST i la Web

Aquests arxius es poden trobar a les carpetes "Laravel - RedditWho API REST" i "Laravel - RedditWho Web" respectivament.

Annex 2. Codi font (extractes)

A2.1. Scripts del bot en Python

Contingut del script **oauth_token.py**. El següent codi és el primer que cal executar, per tal d'aconseguir un *token* de refresc de *Reddit*, amb el que poder fer ús de la seva API.

```
from redditWhoLib import loginData
import praw
import webbrowser

r = praw.Reddit(loginData.APP_UA)

r.set_oauth_app_info(loginData.APP_ID, loginData.APP_SECRET, loginData.APP_URI)

url = r.get_authorize_url('...', loginData.APP_SCOPES, True)
webbrowser.open(url)

app_callback_code = input('Codi d\'autorització: ')
access_information = r.get_access_information(app_callback_code)

print('Refresh Token: %s' % access_information['refresh_token'])
```

Extracte del contingut de la classe **baseDades** del script **utils.py**.

```
class baseDades(object):
    ''' Classe retornada pel mètode dblogin()
        :atributtes: objecte amb dos atributs
        :atype: pymysql.connections.Connection object
        :atype: pymysql.cursors.Cursor object
    '''
    def __init__(self, connection, cursor):
        self.con = connection
        self.cur = cursor

def dblogin():
    ''' Per connectar amb la base de dades
        :return: connexió amb la BBDD
        :rtype: pymysql.connections.Connection object
        :rtype: pymysql.cursors.Cursor object
    '''
    try:
        connection = pymysql.connect(
                     host = loginData.DB_HOST, user = loginData.DB_USER,
                     password = loginData.DB_PASS,
                     db = loginData.DB_NAME,
                     use_unicode = True, charset='utf8mb4')

        cursor = connection.cursor()

        cursor.execute('USE '+ loginData.DB_NAME +';')

...
```

Contingut de la funció **storeExcept**. La utilitzo per guardar en una taula els errors que es produeixen durant l'execució dels scripts. Donat que poden estar corrent durant dies, es necessari guardar aquests missatges d'error per facilitar la identificació de *bugs*.

```
def storeExcept(e, cur, con):
    ''' Guarda l'excepció passada a la base de dades

        :param e: el text amb la descripció de l'excepció
    '''
    try:
        error = re.escape(str(e))
        query = 'INSERT INTO excepts (description) VALUES("{0}")'.format(error)
        cur.execute(query)
        con.commit()
    except pymysql.MySQLError as e:
        return
```

Extracte de la funció **smartinsert** del script **utils.py**. En aquest fragment es pot observar com es pobla el diccionari *postdata*, que a continuació s'utilitza a la *query* de *SQL*, per poblar els diferents valors que es guardaran a la base de dades. També s'observa l'ús de la funció *storeExcept*, en cas de que es produeixi un error.

```
postdata = {
    'idstr': o.id,
    'idint': b36(o.id),
    'title': re.escape(o.title),
    'author': o.authorx,
    'subreddit': o.subreddit.display_name,
    'score': o.score,
    'ups': o.ups,
    'downs': o.downs,
    'num_comments': o.num_comments,
    'is_self': o.is_self,
    'url': o.url,
    'created_utc': o.created_utc,
    'over18': o.over_18
}

try:
    newposts += 1
    query = "INSERT INTO posts VALUES('{idstr}', {idint}, '{title}', '{author}',
'{subreddit}', {score}, {ups}, {downs}, {num_comments}, {is_self}, '{url}',
{created_utc}, {over18})".format(**postdata)
    cur.execute(query)
except pymysql.MySQLError as e:
    text = 'smartinsert:Insert. ID: {0}. Excepció: {1}'.format(o.id, str(e))
    storeExcept(text, cur, con)
    pass
```

El següent fragment és un extracte de la funció **getSubmissions()** del script **redditWhoBot.py**. Realitza la tasca de capturar els noms dels *subreddits* amb més subscriptors, limitats a un nombre concret per la variable TOP_SUB_LIMIT. A continuació, crida a la funció *get_all_posts(),* a la que li passa el nom de cada *subreddit*. Aquesta, al seu torn, extraurà les publicacions de cada *subreddit* passat.

```
select = db.cur.execute('SELECT display_name FROM subreddits ORDER BY subscri-
bers ' +
                        'DESC LIMIT %s;' % TOP_SUB_LIMIT)

if (select == TOP_SUB_LIMIT):
    subNames = db.cur.fetchall()

else:
    raise pymysql.MySQLError('S\'esperaven %d files i se n\'han extret %d'
                             % (TOP_SUB_LIMIT, select))
    return

subList = []
for subName in subNames:
    subList.append(subName[0])

for subreddit in subList:
    (newposts, updates, subsCount) = get_all_posts(subreddit=subreddit, db=db,
r=r, subsCount=subsCount, lower=START_DATE)
    totalNewposts += newposts
    totalUpdates += updates
```

Per últim, un extracte de la funció **get_all_posts()**, també del script **redditWhoBot.py**. Aquest fragment cerca en un interval de temps variable, i captura totes les publicacions fetes al *subreddit*. Si el nombre de publicacions és massa alt, redueix l'interval i si és massa baix, l'incrementa. Quan es captura un nombre de publicacions que es troba dintre dels límits especificats, es passen a la funció *smartinsert()*, que les emmagatzemarà a la base de dades.

```
if itemsfound < (MAX_SUBMISSIONS * 0.75):
    print('Too few results, increasing interval')
    diff = (1 - (itemsfound / (MAX_SUBMISSIONS * 0.75))) + 1
    diff = min(MAXIMUM_EXPANSION_MULTIPLIER, diff)
    interval = int(interval * diff)
if itemsfound > (MAX_SUBMISSIONS - 1):
    #Intentionally not elif
    print('Too many results, reducing interval')
    interval = int(interval * (0.8 - (0.05*toomany_inarow)))
    upper = lower + interval
    toomany_inarow += 1
else:
    lower = upper
    upper = lower + interval
    toomany_inarow = max(0, toomany_inarow-1)
    (queryNewposts, queryUpdates) = utils.smartinsert(db.con, db.cur, searchre-
sults, MIN_SCORE)
```

Annex 3. Llibreries/Codi extern utilitzats

A3.1. Llibreries Python

Pymysql: https://github.com/PyMySQL/PyMySQL
Llibreria *Python* per facilitar la comunicació amb els gestors de bases de dades *MySQL* i *MariaDB*.

Praw: https://github.com/praw-dev/praw
Llibreria que facilita l'accés i interacció amb *l'API* de *Reddit*.

Inspect
Llibreria estàndard de *Python*, permet observar els continguts dels objectes, separant variables i valors en parells.

Requests
Llibreria estàndard de *Python*, permet treballar amb el protocol HTTP de manera senzilla.

Time
Llibreria estàndard de *Python*, és un conjunt de mètodes i atributs per treballar amb temps.

Datetime
Llibreria estàndard de *Python*, és un conjunt de mètodes i atributs que faciliten el formatat de valors de temps.

Json
Llibreria estàndard de *Python*, és un conjunt de mètodes i atributs que faciliten el treball amb el format *Json*.

Re
Llibreria estàndard de *Python*, és un conjunt de mètodes i atributs que faciliten el treball amb les expressions regulars.

A3.2. Codi extern Python

Timesearch.py:
https://github.com/voussoir/reddit/blob/master/Prawtimestamps/timesearch.py
Codi *Python*, n'he utilitzat i adaptat algunes funcions, per tal de poder explorar els *subreddits* i extreure'n les publicacions, per emmagatzemar-les a la BBDD.

A3.3. Frameworks API i aplicació Web

Laravel: https://laravel.com

Framework de PHP orientat al desenvolupament d'aplicacions web seguint el paradigma model-vista-controlador (MVC). L'he utilitzat tant en el desenvolupament de l'API REST per realitzar consultes a la base de dades, com en el desenvolupament de l'aplicació web que realitza consultes a l'API i mostra els resultats als usuaris.

Guzzle: https://github.com/guzzle/guzzle

Es tracta d'un client PHP HTTP que facilita l'enviament de peticions HTTP. L'utilitzo a l'aplicació web per capturar consultes a l'API.

Annex 4. Captures de pantalla

Figura A4.1. Alta de l'aplicació a Reddit

```
D:\_Grau Multimedia\___Semestre actual (Google Drive)\TFG\Python\redditWhoBot>python subreddits.py
Connectant amb l'API de reddit.
Version 3.4.0 of praw is outdated. Version 4.0.0b4 was released 3 days ago.
Connexió amb l'API correcta. Connectat com a rwhobot.
Connexió amb la base de dades correcta.
Últim subreddit processat: 2sljg. Total processats: 25
Últim subreddit processat: 2qzb6. Total processats: 50
Últim subreddit processat: 2uni5. Total processats: 75
Últim subreddit processat: 2r2o9. Total processats: 100
Últim subreddit processat: 2u4js. Total processats: 125
Últim subreddit processat: 2xp02. Total processats: 150
Últim subreddit processat: 2qofe. Total processats: 175
Últim subreddit processat: 2sa3m. Total processats: 200
Últim subreddit processat: 2sfg5. Total processats: 225
Últim subreddit processat: 2u9wz. Total processats: 250
Últim subreddit processat: 2qn90. Total processats: 275
Últim subreddit processat: 2slu2. Total processats: 300
Últim subreddit processat: 2tgic. Total processats: 325
Últim subreddit processat: 2t0j5. Total processats: 350
Últim subreddit processat: 2rygj. Total processats: 375
Últim subreddit processat: 33td5. Total processats: 400
Últim subreddit processat: 2t0xk. Total processats: 425
Últim subreddit processat: 2qtr8. Total processats: 450
Últim subreddit processat: 354cl. Total processats: 475
Últim subreddit processat: 2qx71. Total processats: 500
Últim subreddit processat: 2qh2b. Total processats: 525
Últim subreddit processat: 2qkhb. Total processats: 550
Últim subreddit processat: 2vm97. Total processats: 575
Últim subreddit processat: 2z02l. Total processats: 600
```

Figura A4.2. Script de captura de subreddits en funcionament.

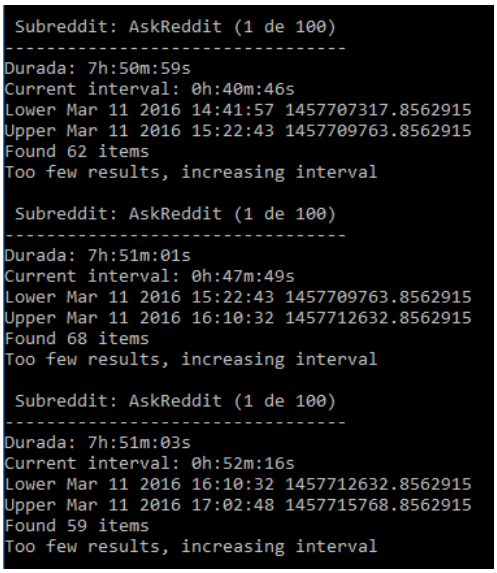

Figura A4.3. Versió beta del bot en funcionament.

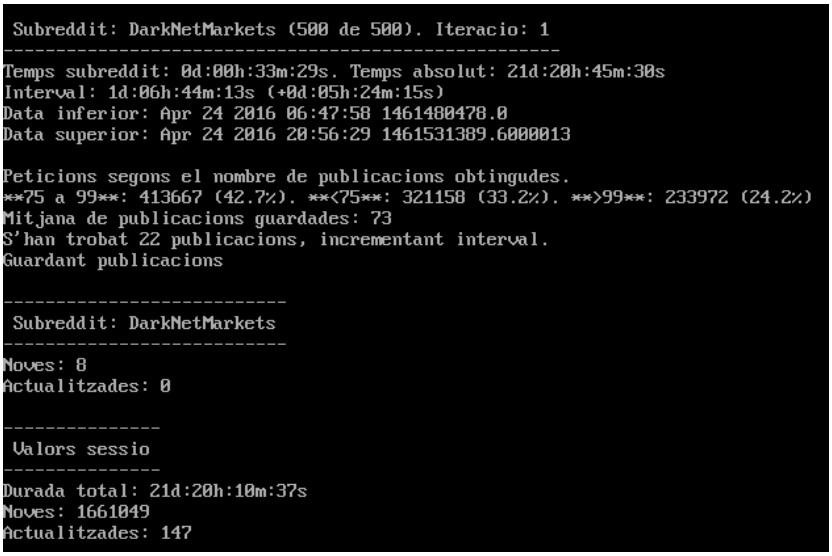

Figura A4.4. Fi primera exploració del bot. Sols 500 subreddits.

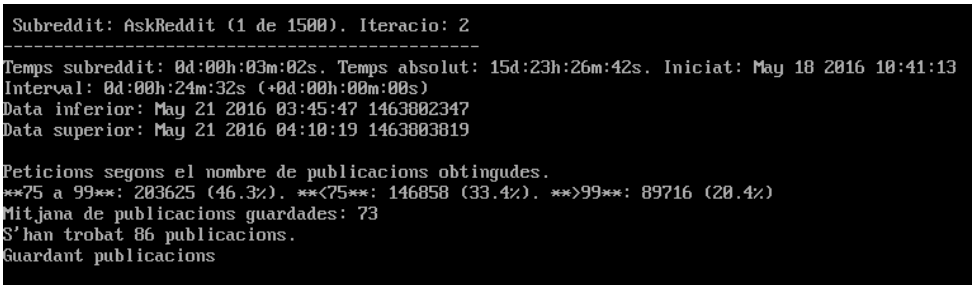

Figura A4.5. Fi segona exploració del bot. 1500 subreddits en total.

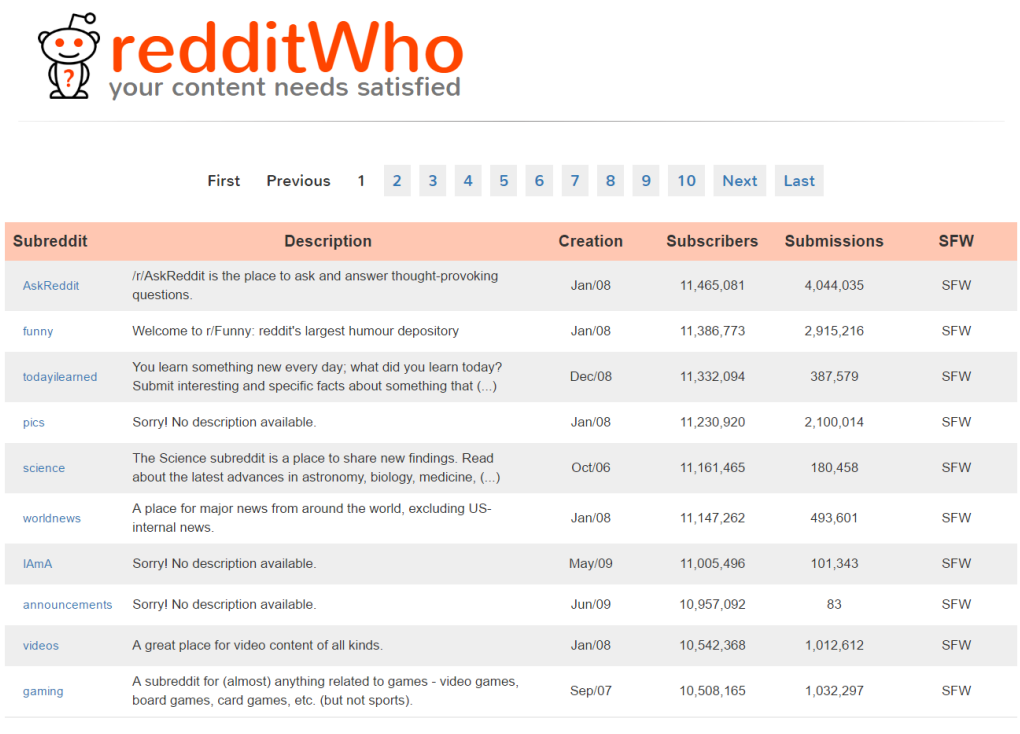

Figura A4.6. Web escriptori.

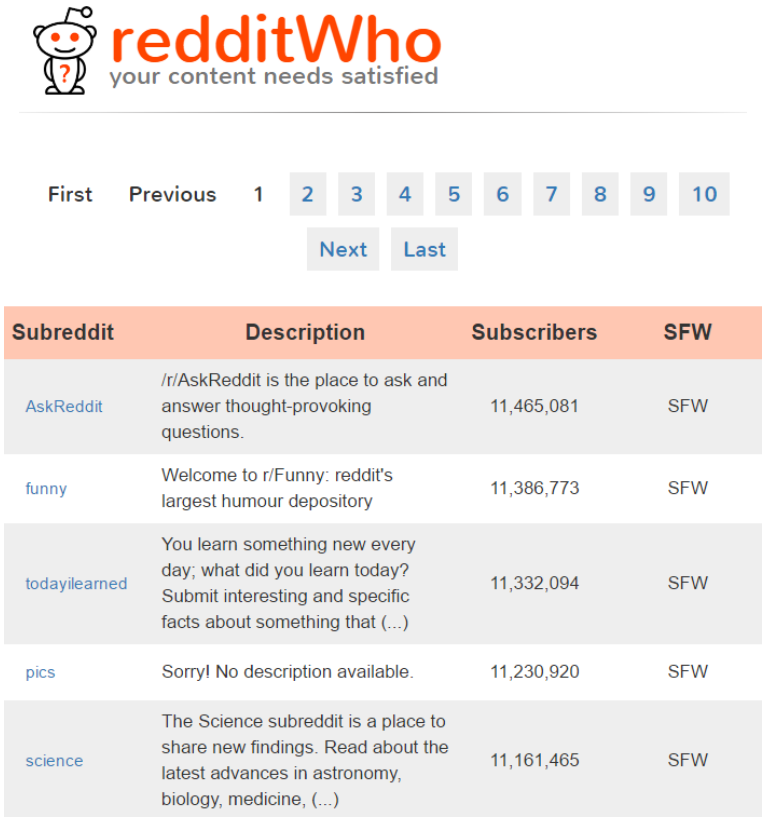

Figura A4.7. Web tauleta.

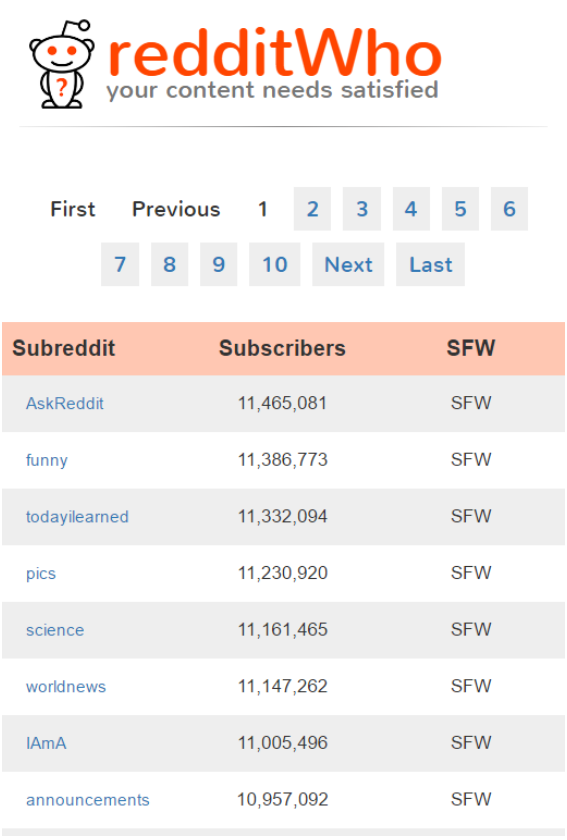

Figura A4.8. Web mòbil apaïsat.

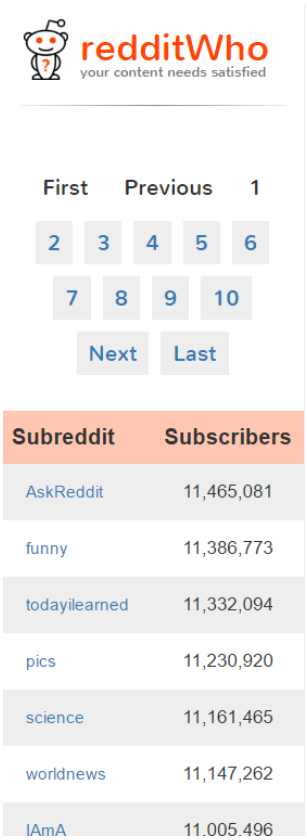

Figura A4.9. Web mòbil vertical.

Annex 5. Resum executiu

Reddit és una pàgina on hi ha centenars de milers de fòrums temàtics, també anomenats *subreddits*. En aquests, els usuaris hi publiquen contingut relacionat amb la temàtica del fòrum i tenen l'opció d'assignar vots positius o negatius a cada publicació, en funció de si els hi agrada el contingut publicat o no.

RedditWho és una pàgina web que incorpora una selecció de *subreddits* o fòrums de la pàgina *Reddit*. Aquests han estat seleccionats pel nombre de subscriptors, en ordre descendent, fins a un màxim de 1500.

D'aquests *subreddits*, es seleccionen les publicacions amb un mínim de 500 vots positius i s'incorporen a la web. D'aquesta manera els usuaris visitants de *RedditWho*, tenen a la seva disposició contingut altament curat i organitzat per matèries.

Aquests poden utilitzar la web amb finalitats àmpliament diverses, com ara la cerca d'idees per generar contingut nou, o simplement consumir contingut prèviament valorat positivament per altres usuaris, d'una forma més fàcil de trobar que amb la interfície de *Reddit*.

Annex 6. Glossari

API Interfície de programació que permet comunicar-se amb un altre programari, a través d'una funcionalitat transparent.

Framework Abstracció d'un llenguatge de programació, que facilita uns fonaments en forma de classes i funcions, que faciliten i acceleren el desenvolupament de programari.

MVC Paradigma Model, Vista, Controlador. És un patró d'arquitectura de programari, que separa les dades i la lògica d'una aplicació, de la interfície d'usuari i el mòdul encarregat de gestionar els esdeveniments i les comunicacions.

Subreddit Categoria dintre de Reddit. P.ex: /r/cars o /r/whitehouse

Superset Superconjunt, una part d'un tot.

Token Un codi alfanumèric utilitzat a l'API de Reddit, per poder identificar l'aplicació o servei, de manera unívoca.

VPS Servidor virtual privat, es tracta d'uns recursos situats en un servidor remot, que es poden utilitzar de manera exclusiva.

Wrapper Conjunt de funcions que abstreuen un programari, per fer-lo més senzill d'utilitzar.

Annex 7. Bibliografia (APA)

Stack Overflow. Retrieved from http://stackoverflow.com/. Múltiples consultes.

Krueger, K. Intro to Relational Databases | Udacity. Retrieved March 18, 2016, from https://www.udacity.com/course/intro-to-relational-databases--ud197

Voussoir. (2014, November 14). Writing a reddit bot - 02 - Writing ReplyBot (UPDATED). Retrieved March 23, 2016, from https://www.youtube.com/watch?v=keiATJcZE8g

Voussoir. (2015, July 08). Writing a reddit bot - 03 - OAuth 2. Retrieved March 26, 2016, from https://www.youtube.com/watch?v=Uvxu2efXuiY

Boe, B. PRAW: The Python Reddit Api Wrapper. Retrieved March 23, 2016, from https://praw.readthedocs.org/en/stable/

Sverdlov, E. (2012, June 12). How To Create a New User and Grant Permissions in MySQL | DigitalOcean. Retrieved March 26, 2016, from https://www.digitalocean.com/community/tutorials/how-to-create-a-new-user-and-grant-permissions-in-mysql

Reitz, K. Requests: HTTP for Humans. Retrieved March 27, 2016, from http://docs.python-requests.org/en/master/

Petri, U., & Gutmann, H. PyFormat Using % and .format() for great good! Retrieved April 1, 2016, from https://pyformat.info/

Documentació Python 3. https://docs.python.org/3/

Documentació PyMySQL. https://github.com/PyMySQL/PyMySQL#pymysql

Eva200. (2015, July 3). Download & Install Centmin Mod LEMP stack. Retrieved April 18, 2016, from http://centminmod.com/download.html#method3

Myadmin. (2015, July 30). How to Install Python 3.4.3 on CentOS/RHEL & Fedora. Retrieved April 18, 2016, from http://tecadmin.net/install-python-3-4-on-centos-rhel-fedora/

Peter, D. (2016, January 22). How to Install Multiple Versions of Python (2.7 and 3.5) Without breaking System tools and Create Isolated environment using Virtualenv? Retrieved April 18, 2016, from http://techglimpse.com/python-sandbox-install-virtualenv-linux/

Miller, S. (2014, December 08). How To Create a systemd Service in Linux (CentOS 7). Retrieved April 19, 2016, from https://scottlinux.com/2014/12/08/how-to-create-a-systemd-service-in-linux-centos-7/

Martell, L. (2013, December 9). [CentOS] making a script into a service. Retrieved April 19, 2016, from http://grokbase.com/t/centos/centos/13c999vnp9/making-a-script-into-a-service

Matula, T. Laravel application development cookbook: Over 90 recipes to learn all the key aspects of Laravel, including installation, authentication, testing, and the deployment and integration of third parties in your application.

Bean, M. Laravel 5 essentials: Explore the fundamentals of Laravel, one of the most expressive and robust PHP frameworks available.