TecnoCampus
Escola Superior Politècnica

Centre adscrit a la
upf. Universitat Pompeu Fabra Barcelona

Grau en Enginyeria Informàtica de Gestió i Sistemes d'Informació

CREACIÓ DE DASHBOARDS PER TROBAR FORATS DE CONTINGUT A LA VIQUIPÈDIA: DESENVOLUPAMENT D'EINES DE RECOMANACIÓ D'ARTICLES SOBRE GÈNERE, GEOGRAFIA I MEDICINA PER LES 310 VERSIONS LINGÜÍSTIQUES DE VIQUIPÈDIA

Memòria Final

DANIEL NAVARRO I CORTÉS

TUTOR: DR. MARC MIQUEL RIBÉ

2020-2021

TecnoCampus
Mataró-Maresme

Dedicatòria

Als meus pares i als meus avis, que m'ho han donat tot, s'han sacrificat perquè gaudeixi del que ells no van poder i m'han ensenyat a ser millor persona.

Agraïments

Aquest treball no hagués estat possible si no hagués arribat fins a l'etapa final d'aquest pedregós camí que és la Doble Titulació en Enginyeria Informàtica i Disseny i Producció de Videojocs. Per tant, vull donar les gràcies als meus amics i companys de carrera que ens hem ajudat i somrigut davant l'adversitat, a Passerells per ser família, a la Carla per ser un far a la foscor, als docents que han fet de la universitat una segona casa i especialment al meu tutor de TFG i professor el Dr. Marc Miquel, que m'ha apropat i guiat pel món acadèmic i ha cregut en mi fins al final del projecte.

Abstract

Wikipedia technical specifications regarding tool development have been studied along some external tools, which have been taken as referents in order to develop a product to offer suggestions to its editors. Hence, helping Wikipedians bridge the existent content gaps inside the encyclopaedia (the content imbalances across different language editions regarding diversity in gender, culture, language, etc.). The resulting product, available at wdo-dev.wmcloud.org, has been an extension of the Wikipedia Diversity Observatory, a project aimed to bridge those content gaps.

Resum

S'han estudiat les característiques tècniques de la Viquipèdia referents al desenvolupament d'eines, a més d'algunes eines externes usades com a referents, amb l'objectiu de desenvolupar un producte que ofereixi suggeriments als seus editors. Conseqüentment, ajudant els Viquipedistes a omplir els forats de contingut existents a l'enciclopèdia (els desequilibris de contingut a través de les diferents edicions lingüístiques referents a diversitat de gènere, cultura, llengua, etc.). El producte resultant, disponible a wdo-dev.wmcloud.org, ha sigut una extensió del Wikipedia Diversity Observatory, un projecte creat per eliminar els forats de contingut.

Resumen

Se han estudiado las características técnicas de la Wikipedia referentes al desarrollo de herramientas, además de algunas herramientas externas usadas como referentes, con el objetivo de desarrollar un producto que ofrezca sugerencias a sus editores. Consecuentemente, ayudando a los Wikipedistas a llenar los agujeros o brechas de contenido existentes en la enciclopedia (los desequilibrios de contenido a través de las diferentes ediciones lingüísticas referentes a la diversidad de género, cultura, lengua, etc.). El producto resultante, disponible en wdo-dev.wmcloud.org, ha sido una extensión del Wikipedia Diversity Observatory, un proyecto creado para eliminar los agujeros de contenido.

Índex

Índex de figures

Índex de taules

Glossari de termes

BD	Base de dades
CCC	Cultural Content Context (Contingut de context Cultural)
CSV	Comma-Separated Values (Valors separats per comes)
DF	DataFrame (Estructura de dades de la llibreria Pandas)
ESUPT	Escola Superior Politècnica – Tecnocampus
GB	Gigabyte
HCI	Human – Computer Interaction (Interacció Persona-Ordinador)
KB	Quilobyte
KPI	Key Performance Indicator (Indicador clau de rendiment)
LPOV	Linguistic Point of View (Punt de vista lingüístic)
Moviment	El Moviment Wikimedia[1]
NPOV o PVN	Neutral Point of View (Punt de vista neutral)
RAM	Random Access Memory (Memòria d'accés aleatori)
TFG	Treball de Fi de Grau
WDO	Wikipedia Diversity Observatory
WDQS	Wikidata Query Service – Servei de consultes de Wikidata
WMCS	Wikimedia Cloud Services

[1] https://ca.wikipedia.org/wiki/Moviment_Wikimedia

1. Introducció

La Viquipèdia és el tretzè lloc web més visitat del món i el setè a Espanya [1] i el **projecte multilingüe més gran del món** amb més de 300 edicions lingüístiques[2]. A més de ser un dels objectes digitals més estudiats gràcies a les seves característiques [2], es tracta d'un projecte que es tira endavant mitjançant voluntaris, de forma totalment transparent i sense ànim de lucre gràcies al Moviment Wikimedia. El moviment Wikimedia és la comunitat global de col·laboradors dels projectes Wikimedia. Originalment creat al voltant de la comunitat de Viquipèdia, s'ha anat estenent a altres projectes com Viquillibres, Wikidata o Viquinotícies, que reben el suport de diferents ONG, notablement la Fundació Wikimedia, nexe d'unió de tots els projectes del Moviment.

La utilitat i l'èxit de la Viquipèdia es pot explicar gràcies a diferents principis o *guidelines* de l'enciclopèdia com el PVN o NPOV en anglès (ha d'estar escrita des d'un punt de vista neutral), la verificabilitat (tota la informació ha d'estar referenciada per fonts fiables), la notabilitat (no tota la informació és vàlida, sinó que ha de generar cert interès al públic) entre d'altres mecanismes de control de qualitat (antivandalisme, correcció gramatical, etc.) realitzat per bots o pels mateixos editors, que assegura que el contingut (generalment) és fiable i de qualitat, i, de no ser així, s'indica al principi de l'article amb diferents rètols, com per exemple el que es veu a la Fig 1.1 referent a la verificabilitat.

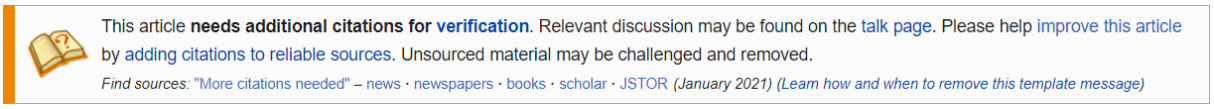


Fig 1.1. Plantilla per demanar més referències als editors per assegurar la verificabilitat. Font: Captura de pantalla de "Template: More citations needed" a la Viquipèdia en anglès, 2021

El que es vol projectar és la creació i desenvolupament d'una eina. Aquesta eina, servirà per donar suport als editors de la Viquipèdia tal que puguin rebre suggeriments d'edició de continguts de temàtiques relacionades principalment amb la diversitat. Es vol que permeti

[2] https://meta.wikimedia.org/wiki/List_of_Wikipedias

l'editor fer-se una idea de l'estat d'una problemàtica concreta dins l'enciclopèdia, mitjançant *dashboards* i estadístiques, expressades a través de l'eina, en un portal web. Per tant, l'objectiu és desenvolupar eines que permetin omplir forats de contingut més fàcilment gràcies a la seva identificació.

Un forat de contingut es dona quan informació important sobre un tema manca, està incompleta, esbiaixada o inaccessible d'alguna manera als lectors [3]. Aquests forats poden ser la infrarepresentació de la diversitat de la gent, de les localitzacions i cultures del món, etc. Per exemple, forats de gènere (percentatge més alt de biografies d'homes que de dones) com es pot veure a la Fig 4.7.

La utilitat del projecte es basa en la millora de l'enciclopèdia i apropar-se a "la suma del coneixement humà", que és d'interès comú, ja que, per exemple, la Viquipèdia s'utilitza com a font d'informació per una part important de la societat, inclòs l'àmbit mèdic, on aproximadament del 50 al 70% de practicants junior utilitzen la Viquipèdia com a font d'informació per donar tractament [4]. A més, per aconseguir més diversitat de continguts i lluitar per l'equitat del coneixement, és necessari representar tots els diferents: **1) localitzacions** (entitats geogràfiques), **2) pobles** (característiques com ara gènere, orientació sexual, grups religiosos, grups ètnics i grups indígenes) , **3) conceptes culturals** per a cada grup de persones i llocs, i **4) idiomes** (nacionals, indígenes i de marginats) del món a la Viquipèdia.

El benefici que aporta el producte recau en l'optimització de processos, el filtratge i l'anàlisi de dades. S'analitzen de forma mensual o inclús diària milers de files de bases de dades per tal d'aportar coneixement amagat entre GB de dades sense processar. Aquest fet, permet als usuaris visualitzar l'estat de la diversitat en el contingut de l'enciclopèdia. A més, identifica els forats i permet accedir a la font d'aquests mitjançant senzills enllaços web. Aquest valor es trasllada a l'enciclopèdia gràcies a la detecció, i posterior cobertura per part dels editors, de **forats de contingut** o *content gaps*.

Addicionalment, el producte està ideat per a un grup d'editors concret que treballi amb una o més de les diferents temàtiques que tracta. Aquesta temàtica està extreta arran d'una selecció de grups d'interès del moviment[3]. Els grups d'interès es componen de Grups

[3] https://meta.wikimedia.org/wiki/Wikimedia_movement_affiliates

d'usuaris de Wikimedia o Organitzacions temàtiques i són agrupacions voluntàries d'editors que amb projectes comuns i que poden abordar el contingut d'un tema concret, per exemple el *Wikimedia Community User Group Math*, sobre matemàtiques o l'agrupació de Wikimedistes Kurds que acapara editors del Kurdistan.

Abans d'acabar la introducció, és adient explicar com s'organitza el treball. Aquest s'organitza mitjançant capítols, apartats, subapartats i de vegades divisions dels subapartats. Primer s'explica el marc teòric en que s'emmarca el projecte, entenent els conceptes necessaris per al desenvolupament, seguit dels objectius i l'abast. Després s'analitzen els referents i es desglossa la metodologia seguida tant per la documentació com pel desenvolupament tecnològic. Un cop definit el projecte, s'explica com ha sigut el desenvolupament del producte i les decisions preses. Finalment, el treball conclou en l'apartat de Conclusions i de Possibles ampliacions, acabant amb la bibliografia .

2. Marc Teòric

L'objectiu de la Viquipèdia és el d'aportar "**la suma del tot el coneixement humà**" [5] disponible per a qualsevol, de forma gratuïta i oberta.

Per a dur a terme aquesta tasca colossal, és necessari que hi intervingui o bé tota la població humana, o bé una part considerable que reculli el coneixement de la resta. Donat que la primera suposició ara per ara és impossible a causa de la manca d'accés a Internet de forma universal i la manca d'alfabetització universal, cal que aquesta part considerable tingui accés a eines per a poder recopilar, discutir, publicar i editar informació (entre d'altres) per a fer la seva tasca, en resum, més fàcil i lleugera.

Què és una eina a Wikimedia?

Les eines són aplicacions de software, aplicacions webs, dispositius i bots que ajuden a la gent que treballa en projectes Wikimedia. Les eines poden fer varis tipus de tasques com ajudar als editors a descobrir tasques ràpides a fer, fer edicions automàtiques, visualitzar dades, extreure metadades i més [6].

2.1. Eines

En aquest subapartat es tracten les característiques de l'espai d'eines del moviment, els tipus d'eines, els seus propòsits i les seves diferències.

2.1.1. Espai d'eines del Moviment Wikimedia: Toolforge

Toolforge és un entorn de *hosting*. Toolforge facilita a l'usuari dur a terme analítiques, administrar bots, córrer serveis web i crear eines [6].

Per a crear una eina o *tool* en anglès o col·laborar en el seu manteniment, es necessita una *Tool Account,* un compte grupal associat a una sola eina, que proporciona accés a:

- Un directori d'emmagatzematge compartit.
- Poder córrer un *web service.*[4]

[4] Col·lecció de protocols i estàndards per intercanviar dades entre aplicacions.

- Credencials per accedir a les rèpliques de les bases de dades en producció.
- Accés a la xarxa de computació o *Grid Engine*[5] per assegurar que una tasca té suficients recursos per executar-se correctament.
- Les credencials i un *namespace*[6] per córrer *containers*[7] a un clúster de Kubernetes.

Cal remarcar que totes aquestes avantatges es poden gaudir de forma **totalment gratuïta**.

Toolforge està estructurat en quatre parts principals:

- *Bastion hosts*, per fer *login* i accedir a les eines de manera interactiva.
- *The grid*, permet als usuaris entrar tasques a una cua, i el sistema troba un *host* per executar-les, ja sigui de manera síncrona o asíncrona, continua o només un cop.
- *The web clúster*, amb un Proxy web, que suporta SSL i és obert a internet.
- *The databases*, Toolforge suporta dos sets de BD, les rèpliques i les generades per usuaris.

Dins de Toolforge es poden usar diferents fonts d'informació relacionades amb el moviment, les ja mencionades rèpliques i els *dumps*, a més de BD generades per altres usuaris.

2.1.1.1. Dumps

Dump vol dir abocador en anglès, ja que conceptualment, els *dumps* són bases de dades, en repositoris grans, com és el cas, de Big Data, on s'hi aboquen dades generades sense cap tipus de filtre previ.

Els *dumps* dels projectes Wikimedia són només de lectura, però es poden descarregar per treballar amb ells o copiar els arxius al directori de l'eina. Wikimedia proporciona *dumps* públics del contingut de totes les wikis i dades relacionades com índexs i *mappings* d'URL curtes. Des de Meta-Wiki fan notar que aquests *dumps* no són còpies de seguretat, no són

[5] https://wikitech.wikimedia.org/wiki/Help:Toolforge/Grid
[6] Contenidor abstracte en el que un grup d'un o més identificadors únics pot existir.
[7] Paquet de software contenent tot el necessari per a córrer una aplicació (codi, llibreries, valors per defecte, etc.)

consistents i no són complets. Les dades que no s'aboquen són les privades, com contrasenyes, e-mails, preferències, dades que permetin identificar usuaris, etc.

Per reduir cert nivell d'abstracció i exemplificar els dumps, a la figura Fig 2.1 es poden veure les taules que formen un *dump* d'un projecte (enwiki, cawiki, cawikictionary, etc.) qualsevol[8].

Table	Filename format	Schema documentation	Description
categorylinks	<wikiname>-YYYYMMDD-categorylinks.sql.gz	categorylinks table schema	All categories with number of pages, subcats, files in each
category	<wikiname>-YYYYMMDD-category.sql.gz	category table schema	Page ids and the categories to which they belong
change_tag	<wikiname>-YYYYMMDD-change_tag.sql.gz	change tag table schema	All tags and the log entry, rc or rev to which they were applied
externallinks	<wikiname>-YYYYMMDD-externallinks.sql.gz	externallinks table schema	Page ids and the off-wiki links they contain
flaggedpages *	<wikiname>-YYYYMMDD-flaggedpages.sql.gz	flaggedpages table schema	Page ids and info about their latest stable versions
flaggedrevs *	<wikiname>-YYYYMMDD-flaggedrevs.sql.gz	flaggedrevs table schema	Revision ids and info about how they have been reviewed
geo_tags	<wikiname>-YYYYMMDD-geo_tags.sql.gz	geo_tags table schema	Coordinate info contained in each page
image	<wikiname>-YYYYMMDD-image.sql.gz	image table schema	Information about uploaded files
imagelinks	<wikiname>-YYYYMMDD-imagelinks.sql.gz	image links table schema	Page ids and their links to media files
iwlinks	<wikiname>-YYYYMMDD-iwlinks.sql.gz	iwlinks table schema	Page ids and their links to pages on other wikis
langlinks	<wikiname>-YYYYMMDD-langlinks.sql.gz	langlinks table schema	Page ids and the equivalent pages on other wikis
page	<wikiname>-YYYYMMDD-page.sql.gz	page table schema	Page info: namespace, title, current revision, etc.
pagelinks	<wikiname>-YYYYMMDD-pagelinks.sql.gz	pagelinks table schema	Page ids and their links to other pages on this wiki
page_props	<wikiname>-YYYYMMDD-page_props.sql.gz	page props table schema	Page ids and various properties of the page (default sortkey? in hidden categories?)
page_restrictions	<wikiname>-YYYYMMDD-page_restrictions.sql.gz	page restrictions table schema	Info about pages protected from editing or moving
protected_titles	<wikiname>-YYYYMMDD-protected_titles.sql.gz	protected titles table schema	Info about titles for which pages cannot be created
redirect	<wikiname>-YYYYMMDD-redirect.sql.gz	redirect table schema	Pages that are redirects and their targets
sites	<wikiname>-YYYYMMDD-sites.sql.gz	sites table schema	Info about all wikis: language code, wiki type, etc.
site_stats	<wikiname>-YYYYMMDD-site_stats.sql.gz	site stats table schema	Sitewide statistics: page views, total edits, etc.
templatelinks	<wikiname>-YYYYMMDD-templatelinks.sql.gz	templatelinks table schema	Page ids and the templates they contain
user_groups	<wikiname>-YYYYMMDD-user_groups.sql.gz	user groups table schema	User ids and the groups to which they belong (bot, sysop, etc)
wbc_entity_usage	<wikiname>-YYYYMMDD-wbc_entity_usage.sql.gz	wbc entity usage schema	Wikidata entity ids and the pages that use them

Fig 2.1.Taules dels dumps d'un projecte, el seu format i una descripció. Font: Captura de pantalla de *Data dumps/What's available for download* a Meta-Wiki, 2021.

2.1.1.2. Rèpliques

Són BD replicades de les que s'utilitzen en producció, és a dir, en viu, a les diferents versions lingüístiques de la Viquipèdia. Són idèntiques excepte per no contenir dades privades d'usuari (algunes files s'han elidit i algunes columnes s'han fet NULL, depenent de la taula). Les credencials per connectar-se a elles es generen quan es crea el compte de l'eina. Normalment la connexió es fa a través d'SSH. L'esquema de totes les taules de MediaWiki és pot veure a l'enllaç al peu de pàgina[9].

[8] Les taules flaggedpages i flaggedrevs poden no estar disponibles en totes les wikis.
[9] https://www.mediawiki.org/w/index.php?title=Manual:Database_layout/diagram&action=render

2.1.2. Tipus d'eines

En aquest apartat es parla dels tipus d'eines que es poden trobar dins del Moviment. A la Taula 2.1 es pot veure un resum del subapartat.

Eina	**Propòsits**	**Característiques**
Bots	Realitzar tasques repetitives	- Són autònoms - Poden usar heurístiques simples i complicades tècniques de *Machine learning*.
Extensions de MediaWiki	Estendre el funcionament Afegir funcions noves	- MediaWiki té un software propi - Creades tant per desenvolupadors de MediaWiki com per tercers.
Dashboards externs	Visualitzar dades	- Allotjats al web - Requereixen d'un analista que prengui decisions gràcies a les visualitzacions (per sí sols no serveixen de res) - Estratègics, analítics o operacionals

Taula 2.1. Resum dels tipus d'eines, els seus propòsits i les seves característiques. Font: Elaboració pròpia.

2.1.2.1. Bots i Cíborgs

Cap al 2007, quan la Viquipèdia va arribar al seu pic d'edicions, estava rebent més de 180 edicions per minut, per tant cap humà podia revisar tots els canvis. Les comunitats d'usuaris van respondre creant bots i cíborgs [7], que automàticament s'encarreguen de tasques repetitives.

"Els cíborgs són interfícies d'usuari intel·ligents que ajuden els editors a realitzar tasques, combinant el raonament humà amb la alta capacitat computacional. Per exemple, Huggle ajuda a atrapar edicions vandàliques comptabilitzades per milers."[7]. (Vegeu Fig 2.2.).

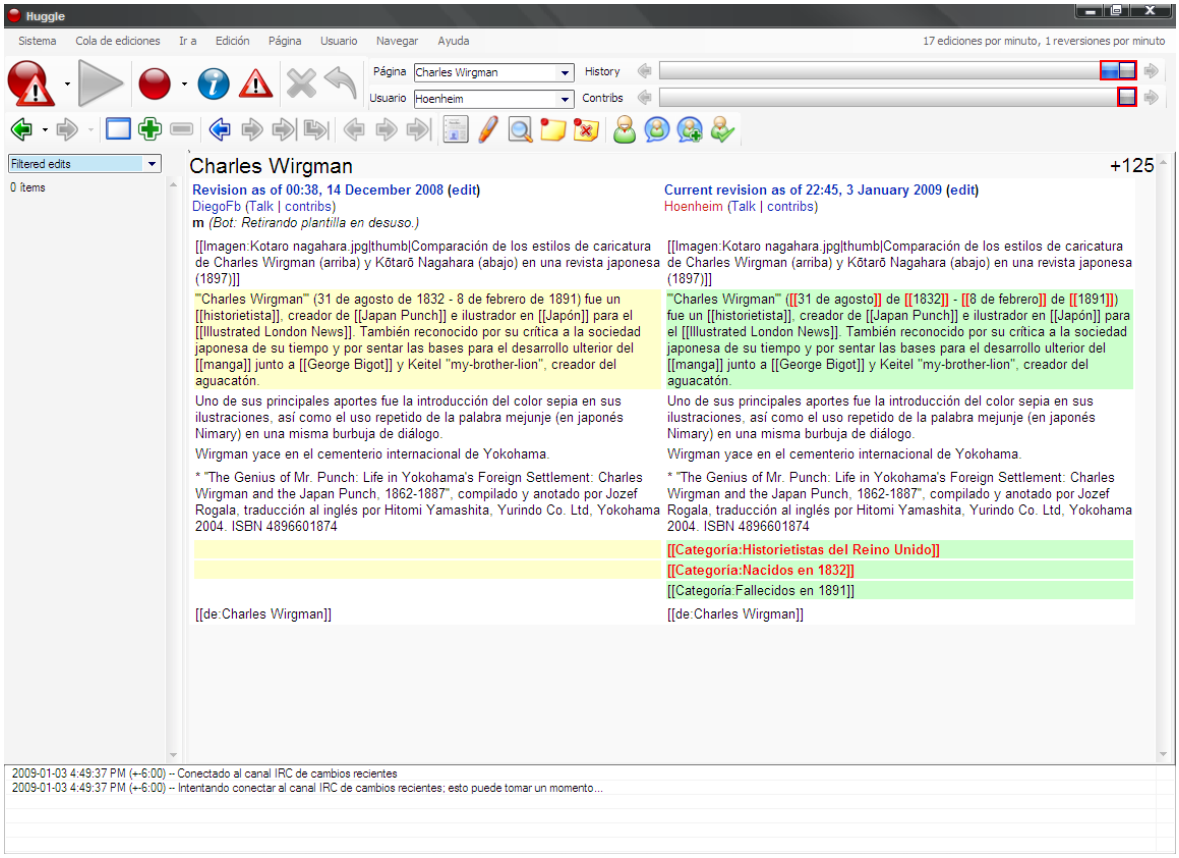


Fig 2.2. Captura de pantalla del programa Huggle creat per Gurch. Font: Huggle Software. Autor: macy, 2007.

A part de poder ajudar moltíssim als editors alleugerant la seva càrrega de "treball"[10], també **poden ser disruptius i dur a terme accions inapropiades**. Per això, "els humans desenvolupen bots, discuteixen la seva aprovació i els mantenen, duent a terme tasques com monitoritzar activitat, fusionar bots similars, separant bots complexos, i apagant bots amb mal funcionament" [8].

2.1.2.2. Tipus de bots

Investigadors del Stevens Institute of Technology, EUA van crear una taxonomia dels rols dels bots a la Viquipèdia, considerant aquests rols com a extensió del rol dels editors que els dissenyen, ja que mostren les necessitats dels seus creadors [8], resumits a la Taula 2.2.

[10] La contribució és totalment voluntària

Rol	Descripció
Generator	Generen articles basats en plantilles predefinides
Fixer	Arreglen errors a articles per mantenir la informació neta i correcta
Connector	Connecten la Viquipèdia amb altres llocs i BD
Tagger	Patrullen articles i etiqueten articles en diferents plantilles i categories
Clerk	Actualitzen estadístiques, documenten status d'usuari, actualitzen pàgines de manteniment i proporcionar alertes d'articles als Viquiprojectes
Archiver	Arxiven discussions tancades i mantenen el contingut arxivat, indexant-lo
Protector	Detecten i regulen comportaments disruptius
Advisor	Proporcionen suggeriments als editors en articles en que els pogués agradar contribuir
Notifier	Entreguen missatges als editors

Taula 2.2. Rols dels bots i funcions associades. Font: Elaboració pròpia a partir de [8]

2.1.2.3. Extensions de MediaWiki

MediaWiki és un software utilitzat per desenes de milers de llocs webs i milers de companyies i organitzacions. És el programa que impulsa la Viquipèdia.

S'utilitza per reunir i organitzar coneixement i fer-lo arribar a la gent. És una eina que s'utilitza per crear wikis (a més de mantenir i impulsar els projectes del moviment), que s'utilitzen tan públicament, com a font de coneixement d'una temàtica concreta com videojocs o pel·lícules tant com a repositoris de documentació per algunes empreses entre d'altres. Les extensions permeten al software ser més avançat i útil per a diferents propòsits.

Algunes extensions són mantingudes per desenvolupadors de Mediawiki, mentre que d'altres és codifiquen gràcies a desenvolupadors aliens, el que provoca que moltes tinguin *bugs*, no siguin compatibles amb d'altres, etc.

Hi ha nou tipus d'extensions, classificades així pel Manual d'extensions de MediaWiki resumides a la Taula 2.3.

Tipus d'extensió	**Descripció**
Parser tags	Estenen el wiki *makup* amb noves funcionalitats, tan simples com processament d'strings com de complexes extraccions d'informació
Parser functions	Són funcions de sintaxi especial de wiki *markup* que poden "interactuar" amb altres wiki elements dins la mateixa pàgina.
Hooks	Permeten que s'executi codi personalitzat quan es duu a terme un esdeveniment (pe. un *login*)
Special pages	Pàgines especials creades pel software sota demanda per a dur a terme funcions específiques
Skins	Permeten personalitzar l'aspecte de MediaWiki
Magic Words	Tècnica de mapeig per a wiki text i ID unics que s'associen a una funció
API	Servei web que permet accedir a funcionalitats wiki com autenticació, cerca, etc.
Page content models	Permet que les wiki pàgines estiguin composades per altres tipus de dades que no siguin wikitext, com JSON.
Authentication	MediaWiki proporciona dos *frameworks* d'autenticació per a millorar la seguretat mitjançant mecanismes d'autenticació personalitzats

Taula 2.3. Resum dels tipus d'extensions de MediaWiki. Font: Elaboració pròpia amb dades de mediawiki.org/wiki/Manual:Extensions.

Per posar un exemple, l'extensió ContactPage[11] proporciona un formulari de contacte (Vegeu Fig 2.3) per a visitants, implementat com a *Special page* i *Hook*.

[11] https://www.mediawiki.org/wiki/Extension:ContactPage

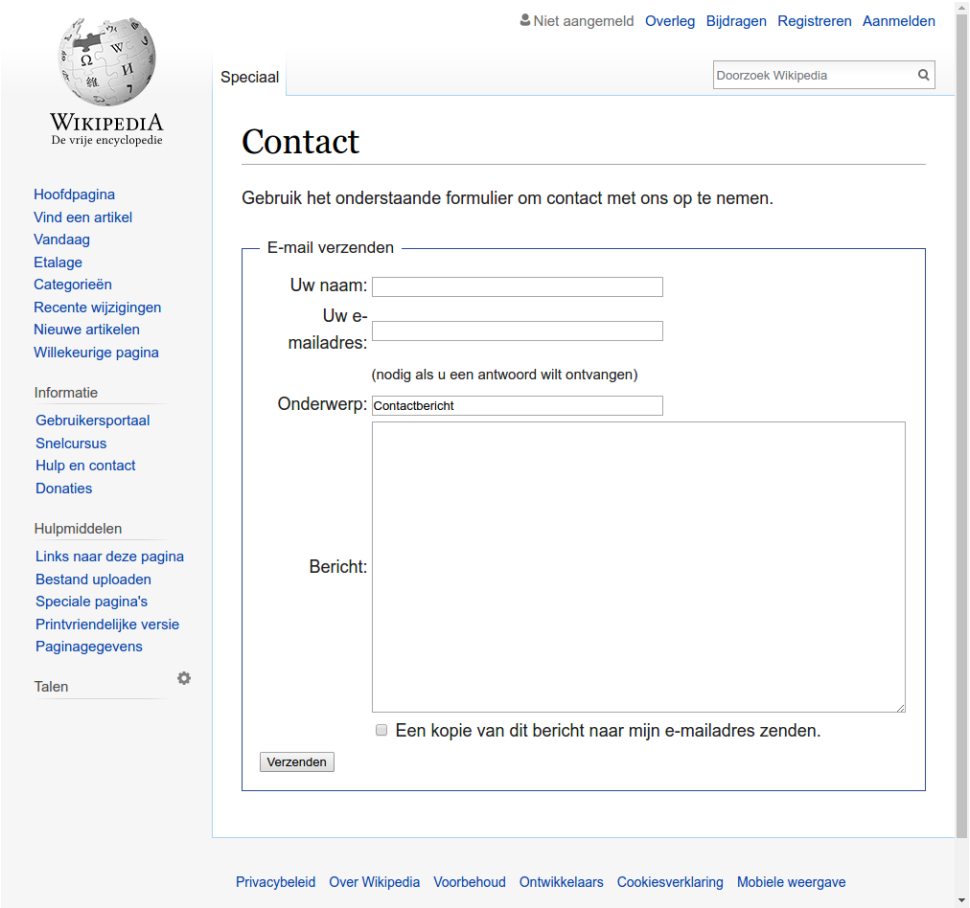


Fig 2.3. Captura de pantalla de l'extensió de MediaWiki ContactPage. Font: nl.wikipedia.org/wiki/Speciaal:Contactpagina Autor: Kghbln, 2017

2.1.2.4. Dashboards externs

Un *dashboard* és un tipus d'interfície gràfica d'usuari que normalment proporciona visualitzacions simplificades de KPI rellevants. Són útils perquè permeten visualitzar rendiment, tendències, ineficiències, etc. En general, estalvien molt de temps respecte a examinar informes un per un, i donen una vista general de tot el sistema [9].

Es poden classificar depenent del seu rol en: estratègics, analítics o operacionals [10]. Per l'objecte del projecte són útils el de tipus analític i estratègic doncs els operacionals no apliquen en el context d'aquest treball.

Els **estratègics** es caracteritzen per ser de més alt nivell i veure oportunitats i el nivell de salut del negoci. Es centren en una "foto" en el temps, presa setmanalment, mensual, anual, etc. Normalment són unidireccionals en el sentit que simplement presenten el que està

passant, i no estan pas pensats per interactuar amb ells per aprofundir a l'anàlisi ans tot el contrari, per donar una idea general [10].

Els **analítics** normalment requereixen més context, com riques comparacions o un historial més extensiu. També es beneficien d'aquestes "imatges" estàtiques, no obstant, visualitzacions més sofisticades són útils per l'analista dedicat, doncs, haurien de suportar interaccions amb les dades (aprofundiment) i poder veure perquè passa, no només què passa [10]. Per exemple, veient una decaiguda dels editors en els últims anys, poder interactuar amb diferents mètriques per poder analitzar i identificar les causes.

2.2. Forats de contingut

Un forat de contingut es dona quan informació important sobre un tema manca, està incompleta, esbiaixada o inaccessible d'alguna manera als lectors [3]. Aquests forats poden ser la infrarepresentació de la diversitat de la gent, de les localitzacions i cultures del món, per exemple, forats de gènere (percentatge més alt de biografies d'homes que de dones) com es pot veure a la Fig 4.7. o culturals, doncs les Viquipèdias més actives tendeixen a representar el context on es parla l'idioma, però fallen a l'hora d'assegurar un mínim de cobertura del context cultural i geogràfic d'altres llengües. A més, hi ha un forat entre edicions lingüístiques de per sí, on alguns articles sovint no es comparteixen o inclús romanen únicament en una edició lingüística [11].

2.2.1. Categories rellevants per la diversitat

Pel que fa a la diversitat dins del contingut de la Viquipèdia, hi ha set categories especialment rellevants, ja que acostumen a estar infrarrepresentades a l'enciclopèdia [12].

2.2.1.1. Forat geogràfic

El forat geogràfic es manifesta principalment per la manca d'articles sobre entitats geogràfiques específiques (ja siguin continents, països, etc.) en la majoria d'edicions lingüístiques.

2.2.1.2. Forat de gènere

Es manifesta principalment per la manca d'articles (biografies) sobre dones en la majoria de les edicions en idioma de Wikipedia en comparació amb articles sobre homes.

2.2.1.3. Forat ètnic

Una ètnia és una categoria de persones que s'identifiquen entre si, generalment sobre la base de presumptes semblances com ara el llenguatge comú, l'ascendència, la història, la societat, la cultura, la nació o el tractament social dins de la seva zona de residència.

2.2.1.4. Forat de grups d'orientació sexual i identitat de gènere

Es manifesta principalment per la manca d'articles (biografies i qualsevol tema) sobre LGTBQ+.

2.2.1.5. Forat de grups religiosos

Es manifesta principalment per la manca d'articles sobre persones que provenen de totes les religions en totes les Viquipèdias. També hi ha un buit en el coneixement sobre els temes relacionats amb totes les religions.

2.2.1.6. Forat cultural

Es manifesta principalment per 1) la manca de representació de CCC en la seva edició lingüística i 2) la manca de compartició o cobertura d'articles en altres edicions lingüístiques que representin el seu context cultural.

2.2.1.7. Forat de llengua

La bretxa lingüística es manifesta en la manca d'una edició lingüística per a totes les llengües que es parlen al món. Segons l'estat de la llengua o el nombre de parlants, entre altres factors, serà més difícil que els parlants es converteixin en col·laboradors (editors). Cal entendre totes les situacions lingüístiques.

2.3. Necessitats d'informació

Per dur a terme el projecte es necessiten dos vessants d'informació, referent a la Viquipèdia i referent a Python i les seves llibreries (Vegeu 6.2.1).

Pel que fa a la **Viquipèdia** com a objecte és necessari entendre:

- Accedir a eines de *hosting* gratuïtes i com utilitzar-les per penjar l'aplicació al web.
- Com sol·licitar l'accés a les BD rèpliques[12] i com consultar-les des de Python.
- Com funciona Meta-Wiki.
- Com funciona Wikidata.
- Com funciona la Fundació Wikimedia.
- Com donar a conèixer l'eina dins del moviment.

Pel que fa al llenguatge de programació **Python** és necessari entendre:

- Sintaxi i mòduls principals de Python.
- Com utilitzar Pandas, NumPy i SciPy, particularment amb datasets molt grans.
- Com utilitzar Plotly i Matplotlib per fer visualitzacions.
- Com utilitzar Dash per crear aplicacions web.
- Com utilitzar SQLite3 amb Python i BD grans.
- Directrius a seguir per complir amb l'estàndard Open Source.
- Com fer un tractament adequat de *cookies* (si s'escau)

2.3.1. Viquipèdia

La Viquipèdia és una enciclopèdia online, i per al seu fi, hi ha una comunitat online d'individus interessats en construir i utilitzar una enciclopèdia d'alta qualitat amb un esperit de respecte mutu [14].

Les seves particularitats flueixen a partir dels "cinc pilars" [15], un resum dels principis fonamentals de la Viquipèdia.

[12] Les BD que hom pot consultar són una rèplica de les que s'utilitzen en producció, doncs hi ha dades privades i el tràfic saturaria el sistema.

- **És una enciclopèdia**, per tant no és una plataforma d'anuncis, un experiment d'anarquia o democràcia, una col·lecció indiscriminada d'informació, etc.
- Cerca el **punt de vista neutral (PVN)**[13], intenta aconseguir que els articles no exagerin un punt de vista específic. Requereix presentar cada punt de vista de forma precisa i dotar de context els articles perquè els lectors entenguin totes les visions i no presentar cap punt de vista com a vertader o millor. Això implica que les opinions, interpretacions i experiències personals dels editors no tenen cabuda. A més cal que tota la informació sigui **verificable**, fet que implica que les fonts dels articles han referenciar fonts confiables i reputades,.
- És **oberta i gratuïta** doncs qualsevol pot usar-la, editar-la o distribuir-la. Cap editor és l'amo de cap article i s'han de respectar sempre els drets d'autor.
- Els editors han de tractar-se amb **respecte i civilitat** entre ells. S'ha d'assumir la bona fe en cas de conflicte, evitar les "guerres d'edicions" i discutir calmadament a les Pàgines de Discussió.
- No té unes **regles inamovibles**. La Viquipèdia consta de polítiques i guies, però no estan escrites en pedra, poden evolucionar i ser interpretades de diverses maneres. A més, s'encoratja perdre la por a cometre errors, doncs totes les versions es guarden i sempre es pot tornar enrere.

2.3.2. Fundació Wikimedia

Wikimedia és un moviment global. La seva missió és fomentar el creixement, desenvolupament i distribució de contingut educatiu multilingüe i gratuït perquè qualsevol persona a escala mundial hi tingui accés.[14]. La Fundació Wikimedia és l'organització sense ànim de lucre que dona suport a Wikipedia i els seus projectes germans.

Els altres projectes germans que fan ús de tecnologies wiki, són: Wiktionary, diccionari de contingut en múltiples llenguatges; Wikiquote, repositori de cites preses de persones famoses o personatges il·lustres; Wikibooks, llibres de text, tutorials, manuals o altres textos pedagògics; Wikisource, biblioteca de textos originals que han estat publicats amb una

[13] https://ca.wikipedia.org/wiki/Viquip%C3%A8dia:Punt_de_vista_neutral
[14] https://www.wikimedia.org/

llicència de lliure accés; Wikispecies, repositori d'espècies que té com a objectiu abastar totes les formes de vida conegudes; Wikinews, font de notícies de tota mena; Wikiversity, plataforma educativa on es poden crear projectes d'aprenentatge, crear contingut didàctic, etc. Wikivoyage, guia de viatges a escala mundial; Commons, repositori per a arxius multimèdia com ara imatges, diagrames, vídeos, etc. MediaWiki, plataforma del programari que és usat per tots els projectes de Wikipedia i per altres Wikis que volen seguir el mateix model; Meta-wiki, lloc dedicat a coordinar els projectes de la fundació; Incubator, projecte on es pot desenvolupar, escriure i provar nous idiomes i accions per a projectes de la fundació; Cloud Services, ecosistema informàtic flexible que potencia la contribució tècnica al món del programari de Wikimedia i Wikidata, centre d'emmagatzematge de dades estructurades per a tots els projectes germans.

Fig 2.4. Els diferents projectes del moviment Wikimedia. Font: Captura de pantalla de Wikimedia.org, 2021

Aquests projectes són el nucli del moviment Wikimedia. Es desenvolupen en col·laboració per usuaris de tot el món que utilitzen programari MediaWiki. Totes les contribucions es publiquen sota una llicència Creative Commons gratuïta, la qual cosa fa possible que qualsevol contingut sigui utilitzat lliurement.

2.3.3. Wikidata

Wikidata és la base de dades secundària, col·laborativa i multilingüe que dona suport a Wikipedia i als altres projectes relacionats que gestiona Wikimedia Foundation [16].

Té com a objectiu proporcionar una font comuna per a certs tipus de dades, com ara informació d'articles de Wikipedia, dates de naixement, localitzacions, matrimoni, edat, entre d'altres.

Les dades que conté estan publicades sota la llicència Creative Commons Public Domain Dedication 1.0, la qual permet la reutilització, la còpia, la modificació, la distribució i l'execució del seu contingut, fins i tot amb finalitats comercials. aquest contingut és proporcionat i mantingut per editors que treballen sobre Wikidata.

Totes les dades de la base de dades són completament multilingües, és a dir, qualsevol nova dada que s'introdueix està automàticament en tots els llenguatges disponibles. A l'ésser dades estructurades, són fàcilment usables i qualsevol usuari o programari pot manipular-los.

2.3.3.1. Estructura de dades

Wikidata consta d'ítems i de propietats.

En el cas dels ítems, cada un d'ells conté un identificador, una descripció i qualsevol quantitat d'àlies. L'identificador es compon amb una Q seguida d'un nombre i és únic.

En el cas de les propietats, igual que els ítems, també contenen un identificador i una descripció. El seu identificador es compon amb una P seguida d'un nombre únic com es veu a la Fig 2.5.

Item	Property	Value
Q42	P69	Q691283
Douglas Adams	educated at	St John's College

Fig 2.5. Exemple d'Ítem-propietat-valor. Font: Wikidata, 2021

Cada ítem té un conjunt de propietats que el descriuen, alhora que aquestes propietats poden enllaçar a un valor que sigui un segon ítem, tal com es pot veure a la Fig 2.6 amb San Francisco, Califòrnia i *United States*.

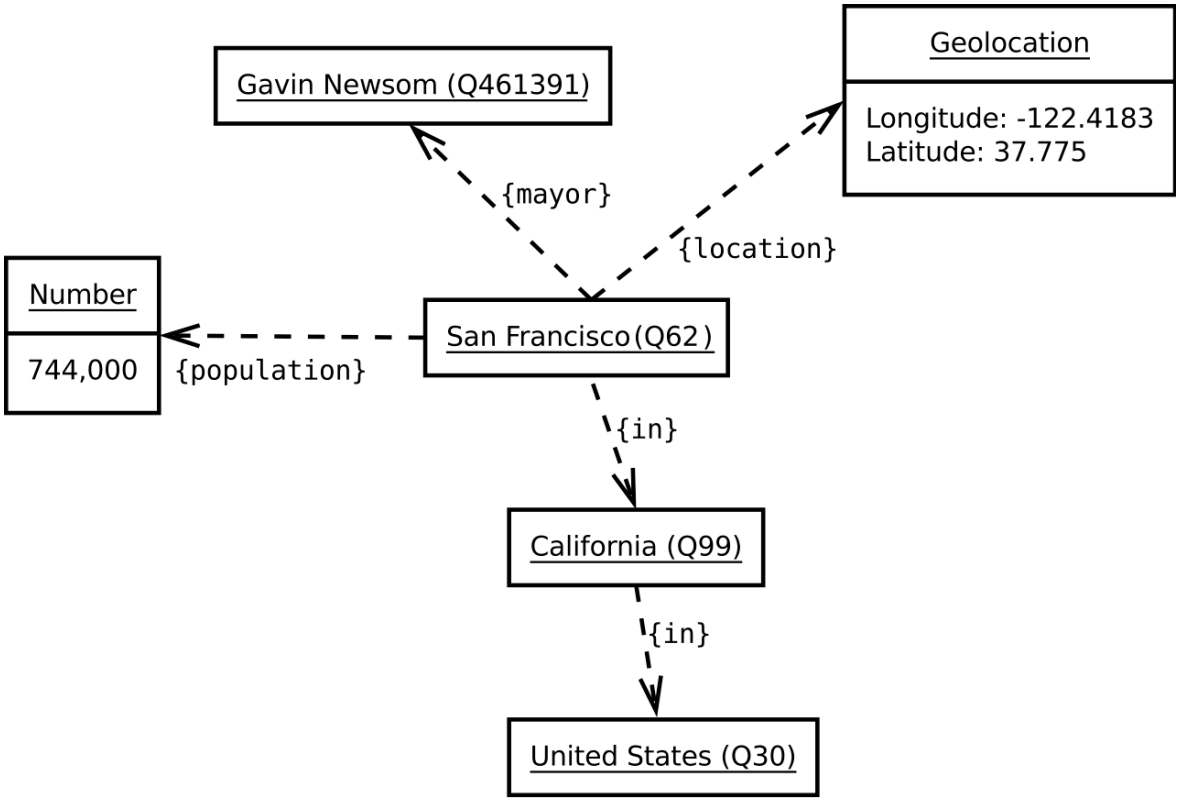


Fig 2.6. Exemple d'Ítems i la interconnexió amb les seves dades. Font: Commons, Jeblad, 2012.

3. Objectius i Abast

En aquest capítol s'estableixen els objectius del projecte i del producte, així com l'abast dels mateixos. Finalment, es parla dels usuaris o clients del producte.

3.1. Objectius

3.1.1. Objectius principals

Objectiu principal 1 (OP1). Desenvolupar eines per processar dades de contingut de la Viquipèdia per poder omplir els forats de contingut que afecten els grups d'interès escollits.

Objectiu principal 2 (OP2). Desenvolupar una infraestructura web per visualitzar les dades processades i facilitar l'accés dels usuaris a la font de les mateixes o la seva interpretació.

3.1.2. Objectius secundaris

Els objectius secundaris relatius a l'OP1 són els següents:

- Donar a conèixer les eines perquè tinguin continuïtat dins del moviment a banda del projectista, entenent continuïtat com a manteniment o col·laboració de tercers en l'eina.
- Desenvolupar software seguint l'estàndard Open Source.
- Conèixer el funcionament pel que fa al desenvolupament d'eines de Meta-Wiki, la web de la comunitat pels projectes de la Fundació Wikimedia, des de la coordinació i documentació fins a la planificació i anàlisi.

Els objectius secundaris relatius a l'OP2 són els següents:

- Treballar amb dades en temps real utilitzant les BD rèpliques de producció.
- Dissenyar una interfície amigable.
- Distribuir l'eina perquè sigui utilitzada per algun dels grups d'interès de (>5% dels editors del grup).

3.2. Abast

L'abast del projecte està ben delimitat pels objectius anteriors, es marca com a objectiu un nombre de tres visualitzacions de temàtica diferent, amb un algorisme de recomanació o suggeriments per a cadascuna.

Pel que fa a l'abast de les dades que es poden obtenir i processar, està delimitat per les polítiques de la Viquipèdia, així i tot, són molt transparents i hom pot disposar d'absolutament totes les dades de contingut i dels usuaris, excepte aquelles que puguin revelar informació sensible com contrasenyes. Les fonts de dades que es poden utilitzar són tant les bases de dades que s'usen en producció (replicades i sense dades sensibles) com els *dumps* o abocadors on es guarden totes les dades, que permeten treballar sense haver d'establir connexió a les BD i per tant consumir recursos.

Per últim, l'abast també s'ha d'encabir en el marc del projecte Wikipedia Data Observatory, doncs el producte estarà allotjat al mateix servidor i el farà créixer mitjançant l'extensió del mateix, aportant o estenent noves funcionalitats. Per tant, el disseny i la implementació del producte també està restringit en certa manera per les limitacions del servidor i l'arquitectura del codi existent, ja que redissenyar tot el codi del WDO no té cabuda ni en l'objecte ni en l'abast temporal d'aquest projecte.

3.3. Producte i client

Els clients són els editors de la Viquipèdia que vulguin participar en les temàtiques escollides, o que tinguin certa sensibilitat amb els forats de contingut. El públic potencial són tots els editors de la Viquipèdia en totes les seves edicions lingüístiques que puguin entendre l'anglès, o en el seu defecte, utilitzar un traductor en línia per poder llegir les instruccions o filtres de cerca del producte.

4. Anàlisi de referents

En aquest capítol és on resideix el punt de partida del projecte, doncs s'analitzen les eines de referència. A la Taula 4.1 es poden veure llistades les conclusions de l'apartat 4.1.

Dashboard	**Elements a destacar**	**Crítiques**
Manypedia	Percentatge de similitud Mètriques utilitzant *outlinks* Precursor Extracció d'imatges dins d'un article	Falta de continuïtat Poca profunditat d'anàlisi Visiblement no atractiu Poca precisió en la traducció
Contropedia	Diferents tipus de visualitzacions i vistes Vista detall amb mètriques i metadades Paràmetres de cerca, acotació i filtratge de resultats Creació de gràfics dins d'altres vistes Barra lateral de context de navegació Extracció de mètriques en temps real	Falta de llegendes Codi de colors no clarificat Gràfics molt poc entenedors en determinades circumstàncies Pocs articles analitzables
GapFinder	Senzill Directe Cerca i suggeriment de text en temps real Redireccionament per editar directament a la Viquipèdia usant l'eina "Content translation".	Manca de filtres d'ordenació d'articles No funciona correctament a tots els navegadors Manca de manteniment

WDO	Visualitzacions complexes Filtres útils Llegendes Enllaços a les dades Gran varietat de visualitzacions i eines Actualment actiu	Visualment poc endegant Temps de càrrega El límit de resultats limita les files que compleixen els filtres de cerca.

Taula 4.1. Elements a destacar i crítiques dels *dashboards* rellevants per al projecte. Font: Elaboració pròpia.

4.1. Dashboards externs rellevants

4.1.1. Manypedia

Aquesta eina permet comparar Linguistic Points of View (LPOV) de diferents comunitats d'edicions lingüístiques i aporta un percentatge de similitud entre les comparades. Permet seleccionar un article en una Viquipèdia concreta i comparar-lo amb l'equivalent en un altre Viquipèdia, però automàticament traduïda a l'idioma origen pel servei de Google Translate. Els creadors clamen que tot i no ser una traducció perfecte, elimina la barrera d'haver d'aprendre les dues llengües per fer estudis entre dues llengües [17].

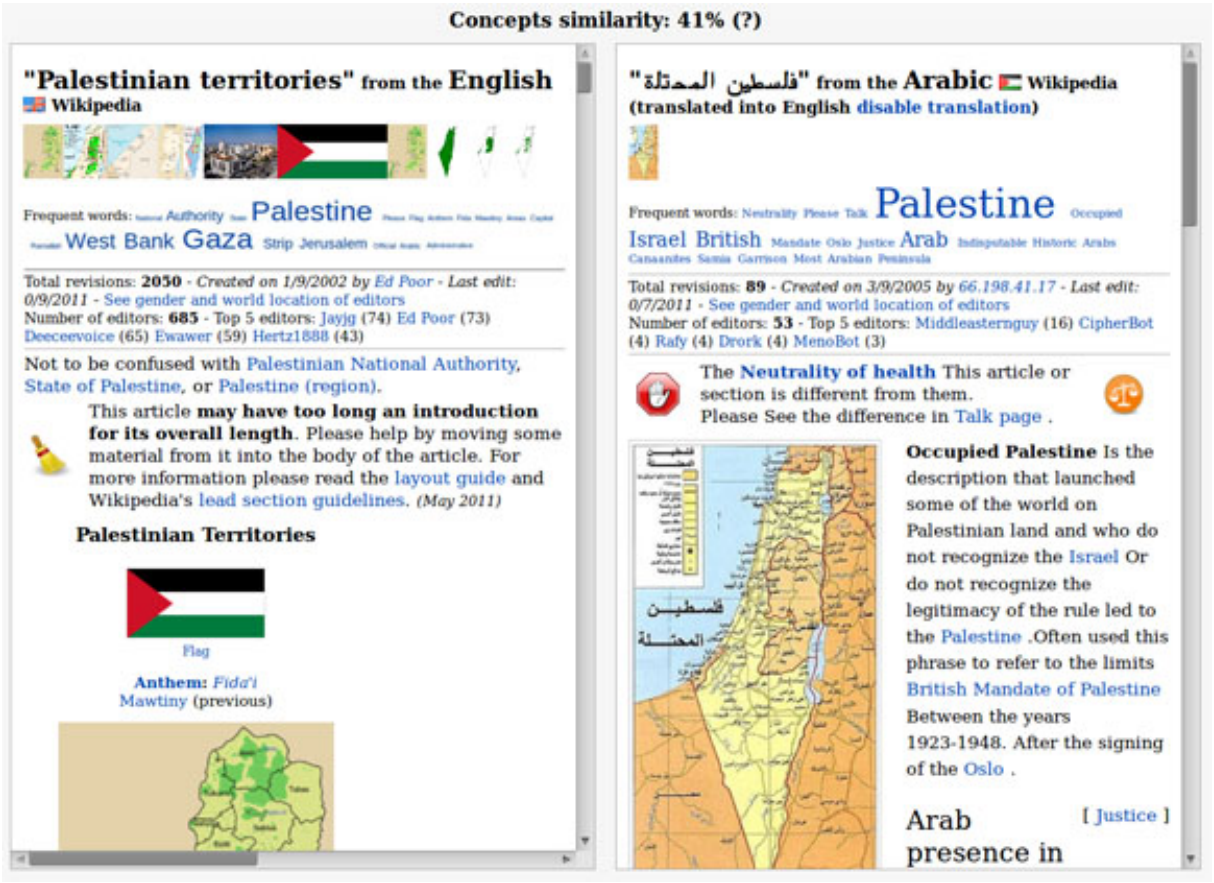


Fig 4.1. Comparació de Manypedia de l'article "Palestinian territories" en la Viquipèdia en anglès i arab. Font: manypedia.com, 2012.

Cada enllaç dins l'article obre una nova "comparació" tal que la navegació pot continuar convenientment dintre de Manypedia [17]. A la capçalera de la comparació, Manypedia mostra informació que pot ajudar al lector a fer-se una idea de les diferencies que hi ha entre ambdues edicions.

Primerament, localitza les fotos incrustades a cada article i les mostra per fer la primera idea dels punts de vista, després genera un núvol de les paraules més freqüents per trobar les diferències textuals. Tot seguit fa una crida a la BD rèpliques mitjançant PHP i Ajax en temps d'execució per mostrar les estadístiques més rellevants de cada article: nombre d'edicions, tenint en compte la diferència de mida de cada comunitat, els editors que han contribuït, que en tractar-se de només un parell podria fer entendre al lector certa infrarepresentació del LPOV o no complir amb el NPOV (biaix). També es mostra el nom

del creador, data de creació de l'article i data d'última edició per donar un sentit d'actualitat [17].

El percentatge de similitud es computa basat en els *outlinks*, articles de la Viquipèdia que apunten a altres articles i la premissa de que "si dos articles del mateix concepte en dos llenguatges defineixen el concepte de manera quasi idèntica, haurien de tenir *outlinks* en gairebé els mateixos conceptes", basat en el "sub-concept diversity index" introduït a Hecth and Gergle (2010).

Malauradament, la seva última actualització al repositori de *Github* va ser fa nou anys i no ha tingut continuïtat fins al dia d'avui i tampoc no està disponible al seu antic lloc web.

4.1.2. Contropedia

Una plataforma per fer anàlisi i visualització en temps real de controvèrsies derivades inevitablement de la creació col·laborativa de contingut. Les mètriques de controvèrsia s'extreuen en temps real dels canals generats per les edicions i els debats sobre articles individuals i grups d'articles relacionats.

"La plataforma està dissenyada amb un enfocament especial a la utilitat de les mètriques i la rellevància pública de les visualitzacions: les mètriques se centren en l'extracció de controvèrsies socials, mentre que les visualitzacions ajuden a donar forma a nous coneixements sobre el tema controvertit." [18].

Contropedia consta de tres tipus de visualitzacions: *layer view*, *dashboard* i *details*.

La ***layer view*** és una vista ràpida de l'article on es ressalten els termes controvertits amb un codi de colors com es veu a la Fig 4.2. A la dreta, la pàgina compta amb una adient visualització de l'article en miniatura, que informa sobre la posició del que s'està veient per pantalla en context amb el total de l'article.

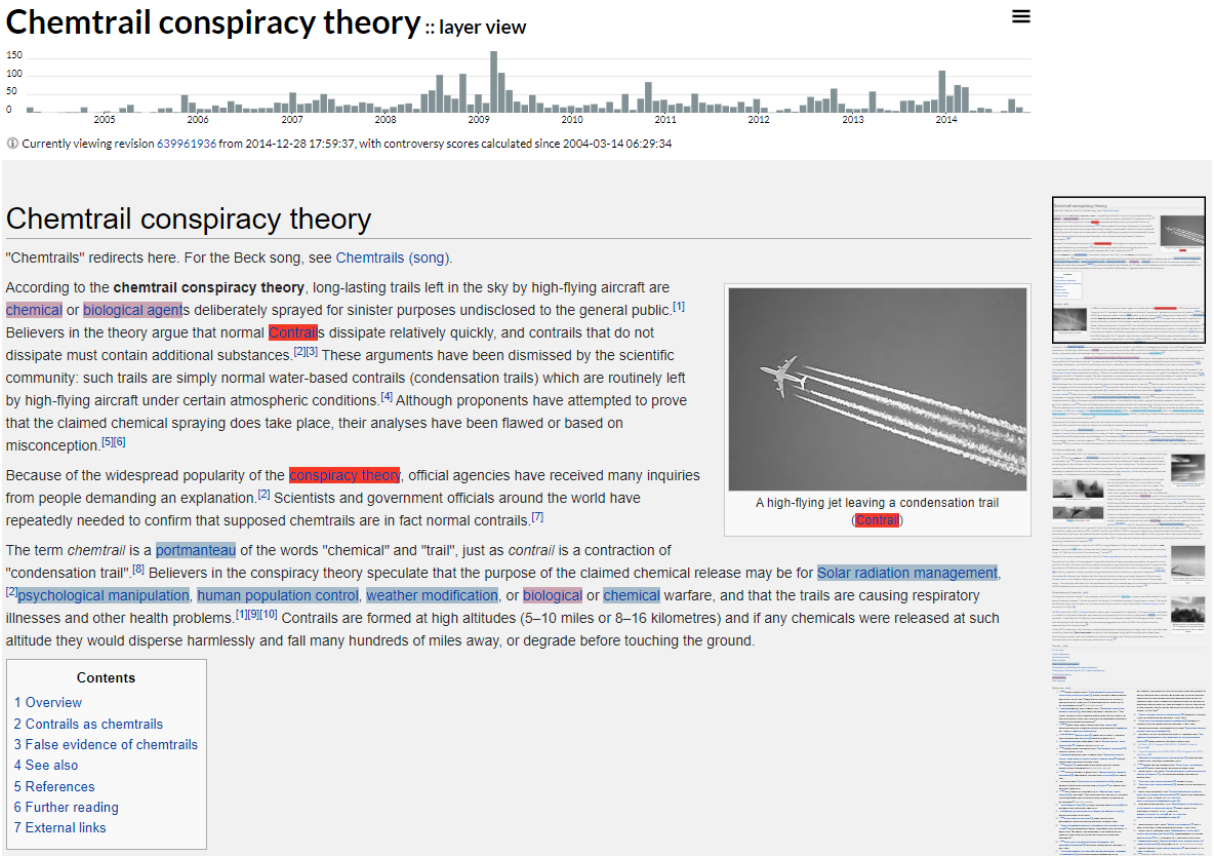
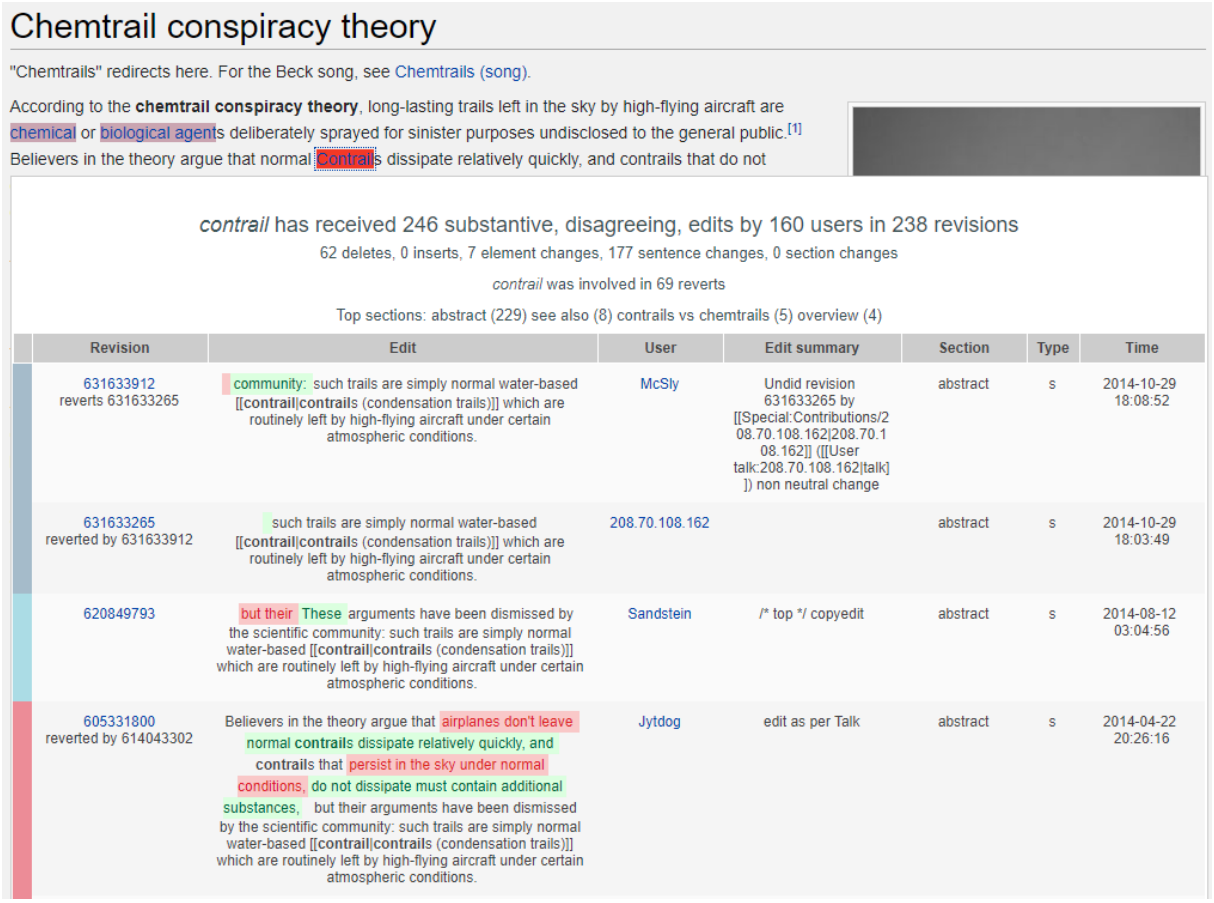


Fig 4.2. Captura de pantalla de la *layer view*: termes controvertits a l'article sobre *Chemtrails*. Les imatges es converteixen a escala de grisos. Font: contropedia.net, 2021.

Es pot clicar qualsevol dels termes i s'obre una finestreta amb els **detalls** de cada terme, on es pot veure tota la informació relacionada amb les edicions i les revisions com es veu a la Fig 4.3. Això inclou ID de la revisió, l'edició, l'usuari[15] que l'ha fet, el comentari d'edició, la secció de l'article on s'ha realitzat l'edició, el tipus i la data i hora de la mateixa. A més, com a resum s'ofereixen les mètriques d'edició del terme: nombre d'edicions "substantives i controversials" el nombre d'usuaris que han participat i el nombre de revisions entre d'altres.

[15] Els usuaris sense compte apareixen com la seva adreça IP pública.



Chemtrail conspiracy theory

"Chemtrails" redirects here. For the Beck song, see Chemtrails (song).

According to the **chemtrail conspiracy theory**, long-lasting trails left in the sky by high-flying aircraft are chemical or biological agents deliberately sprayed for sinister purposes undisclosed to the general public.[1] Believers in the theory argue that normal Contrails dissipate relatively quickly, and contrails that do not

contrail has received 246 substantive, disagreeing, edits by 160 users in 238 revisions

62 deletes, 0 inserts, 7 element changes, 177 sentence changes, 0 section changes

contrail was involved in 69 reverts

Top sections: abstract (229) see also (8) contrails vs chemtrails (5) overview (4)

Revision	Edit	User	Edit summary	Section	Type	Time
631633912 reverts 631633265	community: such trails are simply normal water-based [[**contrail**|**contrail**s (condensation trails)]] which are routinely left by high-flying aircraft under certain atmospheric conditions.	McSly	Undid revision 631633265 by [[Special:Contributions/208.70.108.162|208.70.108.162]] ([[User talk:208.70.108.162|talk]]) non neutral change	abstract	s	2014-10-29 18:08:52
631633265 reverted by 631633912	such trails are simply normal water-based [[**contrail**|**contrail**s (condensation trails)]] which are routinely left by high-flying aircraft under certain atmospheric conditions.	208.70.108.162		abstract	s	2014-10-29 18:03:49
620849793	but their These arguments have been dismissed by the scientific community: such trails are simply normal water-based [[**contrail**|**contrail**s (condensation trails)]] which are routinely left by high-flying aircraft under certain atmospheric conditions.	Sandstein	/* top */ copyedit	abstract	s	2014-08-12 03:04:56
605331800 reverted by 614043302	Believers in the theory argue that airplanes don't leave normal **contrail**s dissipate relatively quickly, and **contrail**s that persist in the sky under normal conditions, do not dissipate must contain additional substances, but their arguments have been dismissed by the scientific community: such trails are simply normal water-based [[**contrail**|**contrail**s (condensation trails)]] which are routinely left by high-flying aircraft under certain atmospheric conditions.	Jytdog	edit as per Talk	abstract	s	2014-04-22 20:26:16

Fig 4.3. Captura de pantalla de la vista detall de "contrail". El color vermell sota la secció "Edit" indica eliminació de text i verd indica inserció. Font: contropedia.net, 2021

L'apartat de ***dashboard*** permet veure de manera més visual l'històric d'activitats d'edició i controvèrsia i el nombre d'usuaris que han participat com es veu a la Fig 4.4. Si es clica a sobre de qualsevol històric, també apareix la vista detall igual que a la *layer view*. A més, si es fa clic sobre la barra de "Users", s'obre un gràfic de que representa la relació entre els editors i les seves contribucions a l'article seqüencialment, tot i que com es veu a la Fig 4.5, pot arribar a ser molt poc il·lustrador i enrevessat per a termes molt controvertits.

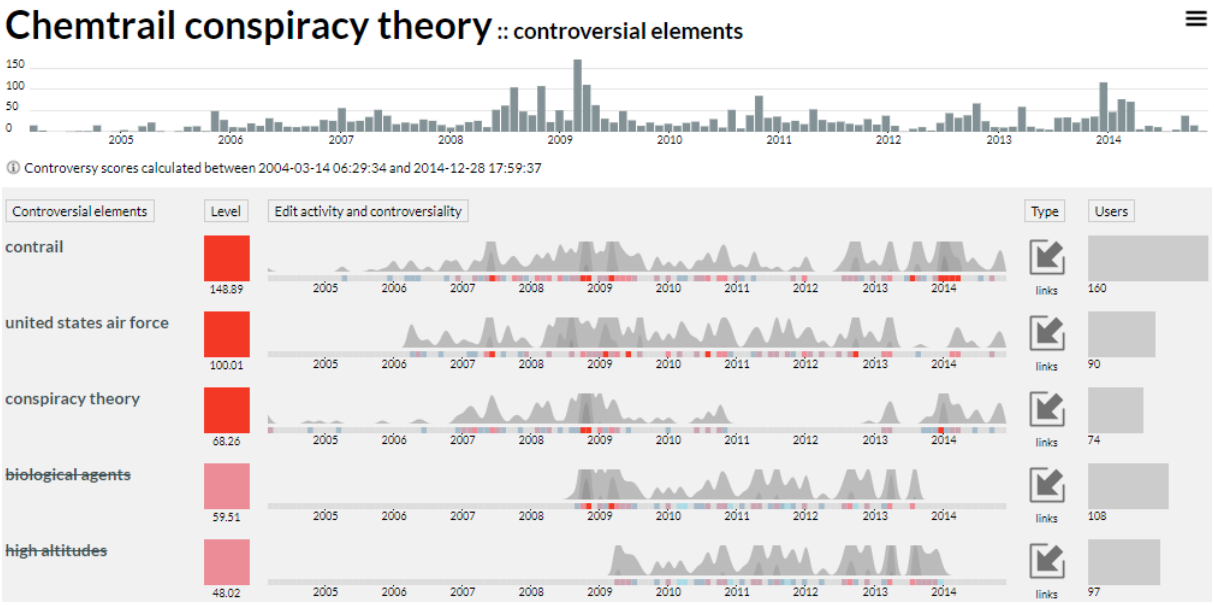


Fig 4.4. Captura de pantalla de la vista Dashboard de l'article sobre Chemtrails. El codi de colors representa l'activitat. Font: contropedia.net, 2021

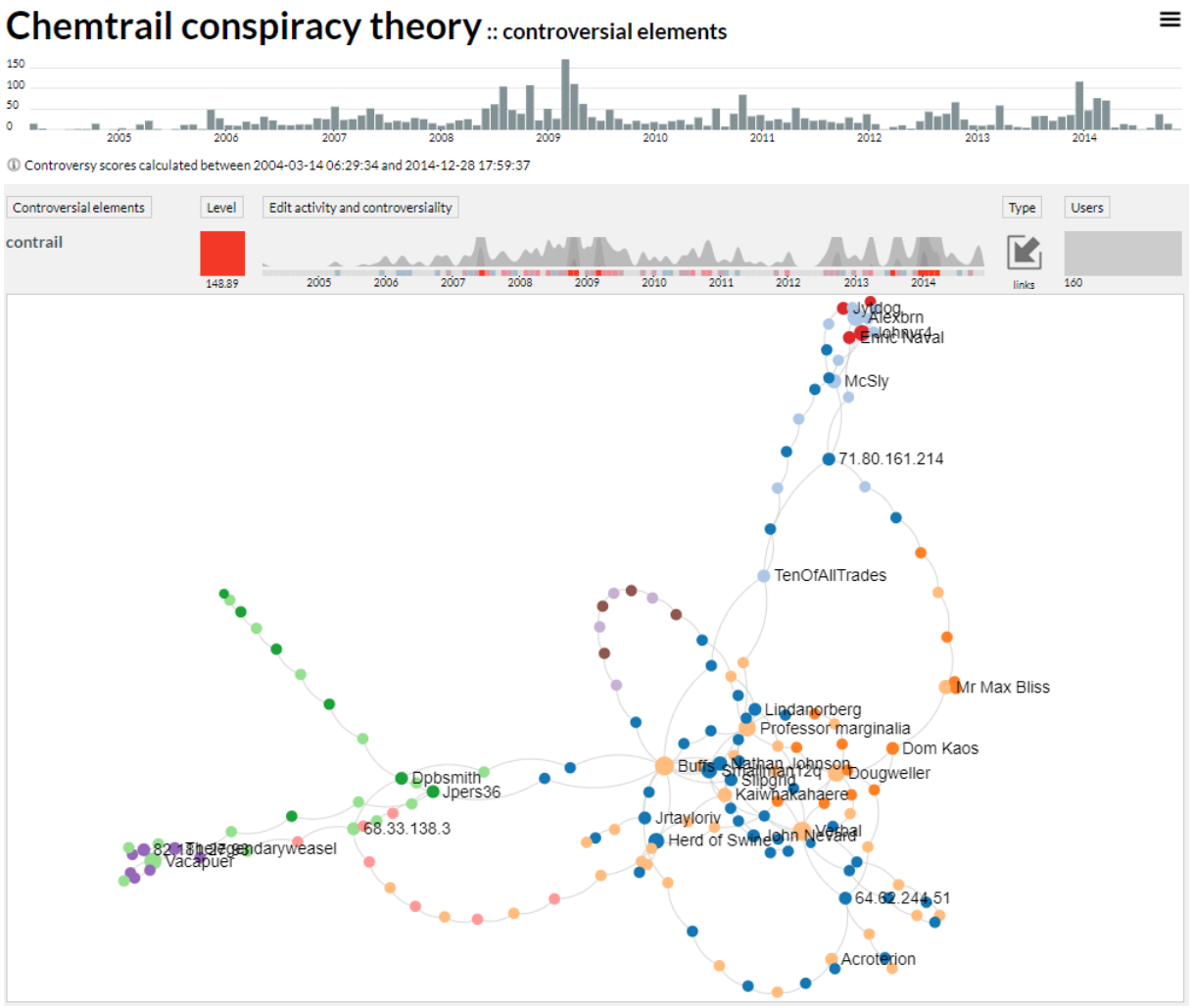


Fig 4.5. Captura de pantalla de la representació gràfica de l'activitat entre editors. El codi de colors representa nivell d'activitat. Font: contropedia.net, 2021

4.1.3. Wikipedia GapFinder

Una eina experimental allotjada a Wikimedia Cloud Services[16] que identifica articles que siguin tendència en una llengua d'origen, però que manquin en una altra llengua destí, a més d'oferir recomanacions en articles relacionats donat un article *seed* o llavor. És una eina molt útil per analitzar els forats de llengua dels articles en tendència i articles rellevants per a l'editor.

[16] https://recommend.wmflabs.org/

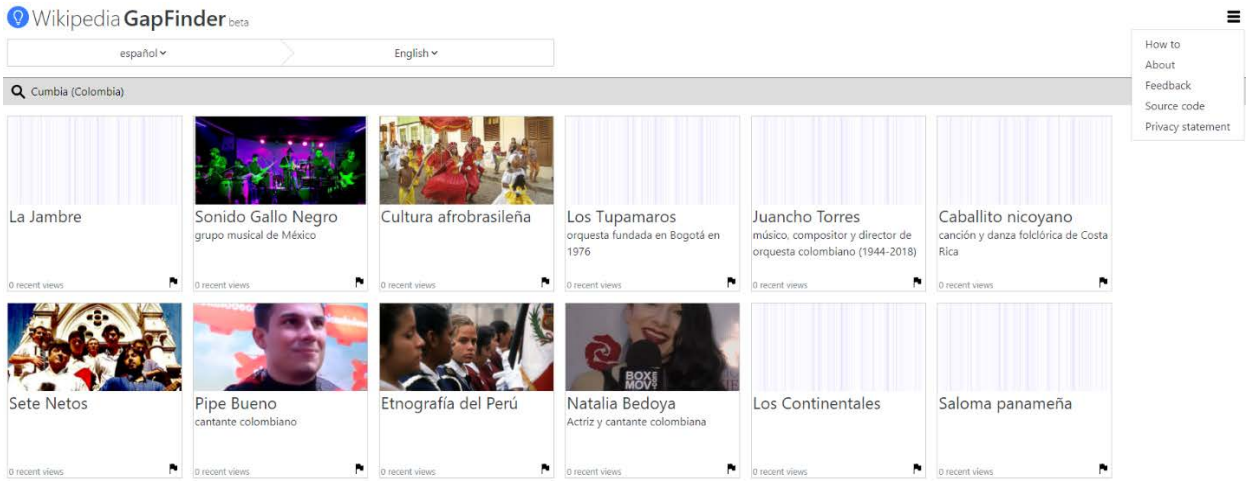


Fig 4.6. Captura de pantalla de la cerca "Cumbia". Espanyol com a Viquipèdia origen i Anglès com a Viquipèdia destí. Font: Wikipedia GapFinder beta, 2021.

Addicionalment, permet redireccionar l'usuari directament a la Viquipèdia per usar l'eina "Content translation" per crear la traducció de l'article o començar-lo des de zero.

Consta d'una API, així que es poden fer cerques sense haver d'utilitzar el navegador. Donada una edició lingüística origen i destí, proporciona els articles d'origen que falten al destí. A més, es poden incloure altres variables com el nombre de recomanacions que es van a buscar, la *seed*, les visites de la pàgina i l'algorisme de cerca (morelike, google, wiki).

Malauradament, no funciona en tots els navegadors, segueix en versió beta i des de 2016 no ha rebut cap actualització.

4.2. Wikipedia Diversity Observatory (WDO): Punt de partida

Com a antecedent de referència i punt de partida del projecte es presenta el Wikipedia Diversity Observatory. La seva visió és alinear el moviment per aconseguir més diversitat de contingut basat en cultura, geografia, gènere, orientació sexual, etnicitat i llengua. Consta de dos elements principals: visualitzacions i eines o *tools*. A més, també proporciona els *datasets* i les BD que utilitza.

Les **visualitzacions** ajuden els usuaris a entendre l'estat de la diversitat dins de les diferents edicions lingüístiques de la Viquipèdia com es veu a l'exemple de la Fig 4.7.

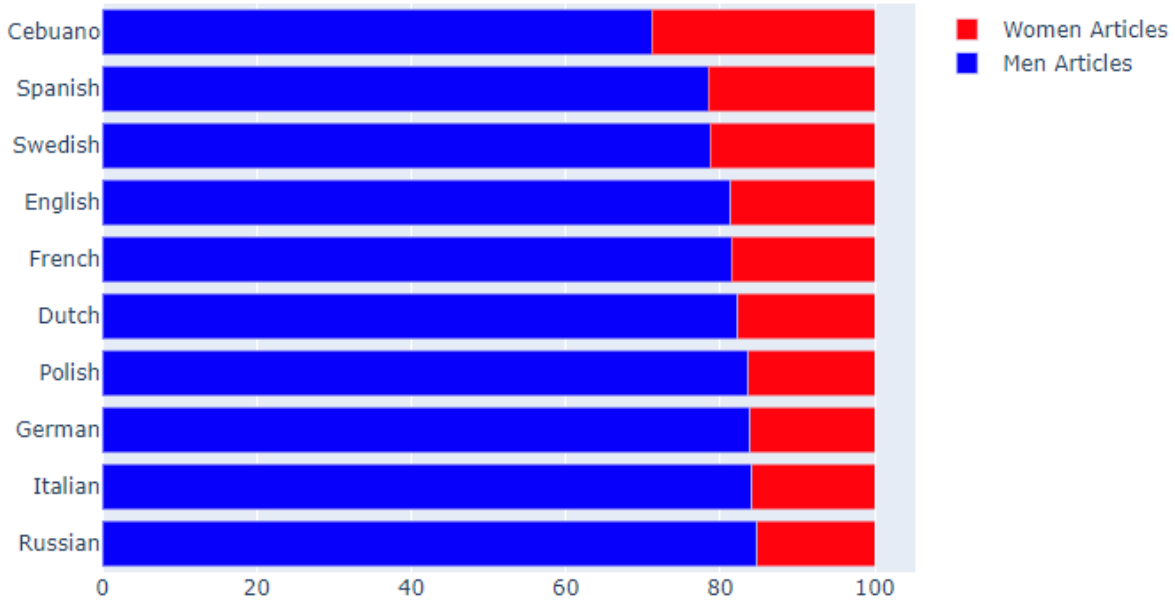


Fig 4.7 Forat de gènere en el Top 10 Viquipèdias a la visualització Gender Gap. Articles de dones en vermell, blau per als d'homes. Font: Wikipedia Diversity Observatory, 2021.

L'objectiu de les **eines** és proporcionar un llistat d'articles que manquen (forats de contingut o *content gaps*) en un context determinat per diferents filtres de cerca interactius com es pot veure a la Fig 4.8.

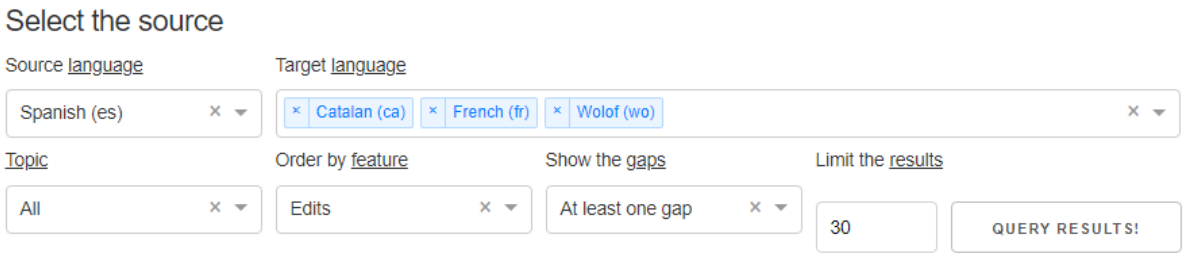


Fig 4.8. Filtres de cerca de la tool " LGBT+ Articles". Font: WDO, 2021

Es pot filtrar per mostrar els articles que: manquen en almenys una llengua destí, que manquin en totes les llengües destí, o tots els articles.

Aquests llistats aporten dades generals útils sobre els articles bo i que no són visualment atractives. A més, compten amb enllaços cap a la font de les dades directament als projectes del moviment com Wikidata o l'edició lingüística pertinent.

S'utilitza com a punt de partida principalment gràcies a la semblança en l'alineació de la missió del WDO amb els **objectius** d'aquest projecte: omplir forats i crear ponts, a la varietat d'**eines** i de **visualitzacions** que té i l' assequible **complexitat** de desenvolupament de l'entorn web i les eines[17]. A més, és un projecte que continua viu en l'actualitat i el tutor del projecte n'és cocreador.

Així, el projecte s'emmarca en una extensió del WDO, aportant visualitzacions o *tools* noves, per tal d'aprofitar la infraestructura i usuaris del WDO i simplificar processos d'implementació i desplegament.

[17] La dificultat recau en el disseny dels datasets i el tractament de dades.

5. Metodologia

En aquest capítol s'expressen els processos i procediments realitzats per assolir els objectius proposats al capítol 3, tant per la redacció de la memòria com pel desenvolupament del producte.

5.1. Producció de la memòria

Per la recerca d'articles acadèmics s'utilitza Google Scholar, Academia i ResearchGate principalment. Les paraules claus a utilitzar principalment són: Wikipedia, Tools, HCI, Editors i Online Communities o una combinació d'aquestes.

Per a l'anàlisi d'articles es llegeix el resum o *abstract* i la introducció, i si es troba adient, s'indaga en l'article sencer en busca d'informació clau relacionada amb l'objecte d'estudi.

Per a aspectes relacionats amb el funcionament de Viquipèdia, s'utilitza la mateixa Viquipèdia i projectes germans.

Per a organitzar la bibliografia, les referències i citacions, s'utilitza el software Zotero, conjuntament amb el complement per Chromium per guardar bibliografia i el complement per Microsoft Word per gestionar citacions i la Bibliografia dins del document.

5.2. Desenvolupament del producte

Les fases de desenvolupament del producte són les següents:

5.2.1. Identificar grups d'interès dins la Viquipèdia per a qui desenvolupar una eina

Identificació mitjançant necessitats de tot el moviment o bé la llista d'afiliats al moviment, concretament les organitzacions temàtiques i els grups d'usuari. Seguint un criteri, primer de tot, d'interès pel projectista, seguit de:

- Utilitat per un actor concret i pel Moviment Wikimedia sencer.
- Reducció de càrrega de treball.

- Valor del descobriment realitzat.

5.2.2. Dissenyar els datasets i els *dashboards* per als grup d'interès

Cal accedir a les fonts de dades, poder fer *queries*, fer *login* a Toolforge i entendre Meta-Wiki per tal de trobar dades valuoses o potencialment valuoses. Gràcies a aquesta vista sobre les dades, es pot valorar quines tractar per tal d'ensenyar quelcom valuós als editors i com ensenyar-ho perquè sigui el més amigable possible, a la vegada que flexible. Un cop se sap el que es vol ensenyar, cal aprofundir en quines taules i variables es necessiten per tal de donar forma al producte.

5.2.3. Desenvolupar l'eina

1. Un cop s'han seleccionat les dades a tractar, cal descarregar-les o obtenir-les en temps real. Els *dumps* es poden descarregar des dels servidors de Wikimedia, i les bases de dades es poden consultar mitjançant crides REST a l'API de Wikimedia o bé utilitzar les bases de dades rèpliques de producció mitjançant la llibreria de MySQL per a Python.
2. Mitjançant la classificació de les dades obtingudes, es creen els conjunts de dades o les bases de dades que s'utilitzen en els punts següents.
3. Utilitzant les dades filtrades de les BD, computar les estadístiques o mètriques i convertir-les en quelcom intel·ligible per als humans utilitzant llibreries de tractament de dades com Pandas.
4. Creació dels *dashboards* o visualitzacions dissenyades, reflectint les dades definitives del pas anterior a l'API web per la visualització i l'anàlisi de dades de manera interactiva. Tanmateix, això implica desenvolupar també la pròpia web per poder ensenyar aquestes visualitzacions. Es duu a terme mitjançant els components HTML de la llibreria Dash i les figures de Plotly, que un cop tenint les dades ordenades amb Pandas, només requereix d'unes senzilles línies de codi per ensenyar-les de forma efectiva.

NOTA: S'assumeix *testing* durant totes les fases del desenvolupament de l'eina a mesura que es van afegint *features* i el desenvolupament avança.

5.2.4. Desplegar l'eina al web i control de qualitat (QA)

Desplegament dels *scripts* al servidor del Wikipedia Diversity Observatory per estalviar costos, facilitar la reproducció i la longevitat del projecte i realitzar tot el *testing* necessari relatiu a la implementació al servidor perquè funcioni com s'espera.

Per a dur a terme la implementació cal, primerament, pujar els *scripts* al servidor FTP, després importar els *scripts* de les APP dins del programa que s'encarrega de la gestió de cada pàgina del web del WDO i crear els elements HTML que permetin accedir a les noves APP amb els seus enllaços corresponents. Per acabar, cal reiniciar aquest programa controlador i esperar a que torni a estar actiu per a que tot internet pugui gaudir, ara sí, de les aplicacions que s'han desenvolupat.

Els missatges que normalment sortirien per la consola de Python per a mostrar al desenvolupador, ara es guarden de forma cronològica en diferents documents en format.*log* per facilitar el testing.

6. Desenvolupament

En aquest capítol es presenta la informació referent al desenvolupament de les eines, el web i el seu desplegament, seguint la metodologia exposada al capítol 5.

Primer, per introduir al lector, es parla sobre el Wikipedia Diversity Observatory i les seves característiques. Seguidament s'expliquen les especificacions tècniques del producte i d'aquest projecte junt amb l'arquitectura del mateix. Aquestes especificacions estan lligades a les del WDO i s'esmenen les decisions tecnològiques que això implica. Per acabar, es detalla el desenvolupament del producte final, començant per l'exposició dels grups d'interès escollits, seguits dels detalls del desenvolupament de cada solució per separat en cada subapartat.

El control de versions del projecte en local es duu a terme mitjançant ***GitHub***, disponible a https://Github.com/Destokado/TFG_Informatica

Els **resultats del treball** es poden veure al web de desenvolupament del WDO, accessible a través del següent web: https://wdo-dev.wmcloud.org/. Es crea un web alternatiu per no destorbar el bon funcionament del web original mentre es fan proves, fins que no s'integri el resultat del treball al web en producció. Per tant, alguns dels *scripts* o *logs* que es mencionen, que pertanyien prèviament al WDO i s'han hagut d'utilitzar i modificar, han sigut duplicats amb l'extensió ***_dev*** per poder treballar sense interferir. Així, en comptes de treballar amb l'*script* que llença el servidor web ***dash_apps.py***, es treballa amb ***dash_apps_dev.py***.

6.1. Característiques i especificacions del WDO

La infraestructura del Wikipedia Diversity Observatory es compon d'un codi font disponible a ***GitHub*** a https://Github.com/marcmiquel/WDO que tracta, recopila i emmagatzema dades, que es poden visualitzar al web https://wdo.wmcloud.org/, que corre en un servidor de Wikimedia Foundation.

El WDO es pot dividir en tres apartats que treballen conjuntament per assolir el seu objectiu de donar visibilitat a l'estat actual de la diversitat de contingut, tal com es descriu al document **Readme.md** del repositori enllaçat anteriorment:

- Dades: Bases de dades de *Wikipedia Diversity* i *Stats*
- Llocs webs: Pàgina web de l'*Observatory* i pàgina a Meta-Wiki
- Recerca: articles i presentacions

6.1.1. Bases de dades

Es crea una base de dades per a cada edició lingüística de la Viquipèdia en què cada article es classifica segons les característiques que poden determinar si pertany a una categoria rellevant per a la diversitat (cultura, gènere, lloc, etc.). Categories com el gènere, l'orientació sexual, la religió o l'origen ètnic són senzilles de determinar, ja que es poden rastrejar mitjançant les relacions semàntiques de Wikidata estructurades com a propietats i ítems com es pot veure a la Fig 2.5, a l'apartat 2.3.3.1.

En canvi, la relació d'un article com a pertanyent als temes relacionats amb la llengua requereix un mètode més sofisticat. En aquest cas, s'utilitza una varietat de funcions basades en el títol de l'article, la categoria i l'estructura del graf d'enllaços, entre d'altres, per etiquetar cada article segons la possible relació amb els territoris on es parla la llengua i amb els pobles que els habiten. A continuació, s'introdueixen en un classificador de *Machine Learning* per obtenir la selecció final d'articles pertanyents a un context cultural i geogràfic d'una llengua. Aquesta col·lecció d'articles s'anomena **Contingut de context cultural (CCC)** i és el grup d'articles d'una edició lingüística de la Viquipèdia relacionada amb el context geogràfic i cultural dels editors (llocs, tradicions, llengua, política, agricultura, biografies, esdeveniments, etc.).

Aquest mètode es construeix mitjançant *Python 3* per manegar les dades, *Sqlite 3* per emmagatzemar-les i *Scikit-learn* per processar-les.

Els conjunts de dades es generen mensualment en format.CSV amb SQLite3.

Per generar la base de dades ***wikipedia_diversity.db*** es creen els següents *scripts*:

- ***wikipedia_diversity.py***, ***content_retrieval.py*** i ***content_selection.py.*** Recuperen les dades dels *dumps* i les BD de Wkimedia, es processen segons criteris i s'introdueixen a la BD.

Pel que fa a la diversitat de contingut, és necessari computar moltes estadístiques basades en CCC, que es realitza mitjançant el següent programa:

- ***stats_generation.py*** calcula aquestes estadístiques i classifica els articles per crear llistes valuoses per a cada edició lingüística. Guarda els resultats a ***stats.db*** de forma mensual per tal de poder crear taules i gràfics amb relació temporal.

6.1.2. Llocs web

Per al desplegament de les eines i visualitzacions del web del WDO s'utilitza el següent *script*:

- ***Dash_apps.py***, que utilitza les llibreries Dash i Plotly per generar les pàgines web utilitzant components **HTML** de forma "pythònica" i els gràfics i visualitzacions respectivament.

Aquest programa importa els *scripts* de cada APP per poder posar en marxa el servidor web en Flask, que corre per sobre del **wsgi (**s'explica al punt següent), amb totes les aplicacions, amb una pàgina cadascuna i els seus enllaços corresponents. Per tal de córrer l'*script* per posar en marxa el web, cal escriure la següent comanda a qualsevol directori ***/srv/wcdo/src_viz:***

sudo systemctl start (o *restart* per reiniciar el procés o *stop* per aturar-lo) *dash_apps*.

Per tal de fer un seguiment del procés des desplegament que mostrarà si està corrent, es pot utilitzar

sudo systemctl status dash_apps

Per llegir els *logs* per comprovar si ha hagut errors i identificar-los, es pot utilitzar

tail -100 errlog

que mostrarà per pantalla les últimes cent línies del *log* d'errors.

6.1.3. Arquitectura

Per poder definir l'arquitectura del desplegament, cal entendre primer els seus components:

6.1.3.1. WSGI, uWSGI, uwsgi

WSGI o Web Server Gateway Interface és una especificació de Python que defineix una interfície estàndard de comunicació entre una aplicació o *framework* i una alplicació/servidor web per simplificar i estandarditzar la comunicació. Bàsicament defineix una interfície API per poder utilitzar sobre altres protocols [19].

uWSGI és un contenidor de servidors d'aplicacions que proporciona un *full stack* per desenvolupar i desplegar aplicacions web i serveis. Permet manegar apps en diferents llenguatges i es comunica amb l'aplicació mitjançant l'especificació WSGI i altres protocols web. Bàsicament tradueix peticions d'un servidor web convencional a un format que l'aplicació pot processar.

Uwsgi és un protocol ràpid binari implementat pel servidor uWSGI per a comunicar-se amb un servidor més ric en funcionalitat.

6.1.3.2. NGINX

Nginx és un servidor web Open Source que actua com a intermediari entre el servidor uWSGI i el client. Compta amb funcionalitats de:

- ***Reverse Proxy***: En un Proxy normal, el servidor no te cap coneixement de quin client li ha enviat la petició, en canvi, en el *reverse*, un client pot estar interactuant amb diferents servidors al mateix temps, sense tenir cap coneixement, i sense que es noti funcionalment. No és necessari que hi hagi més d'un servidor.
- ***Load Balancer***: Complementa el *reverse Proxy* en la mesura que permet distribuir la càrrega de treball de forma eficient en cas que hi hagi múltiples servidors.
- ***Stateless Application***: L'aplicació no guarda dades del client generades en cada sessió, sinó que és el client qui guarda la informació de la seva sessió, i la tramet a les peticions en forma de *token* quan sigui necessari.

Gràcies a aquestes funcionalitats, permet escalabilitat horitzontal de forma fàcil.

6.1.3.3. Flask/Dash Application

Flask és un microframework, ja que no disposa de capa d'abstracció de BD ni autenticació, no obstant es poden afegir mitjançant extensions. Sobre aquest microframework, Dash

construeix les diferents apps en Python. Dit d'una altra manera, Dash està escrit a sobre de Flask i utilitza aquest com a component d'enrutament web.

6.1.3.4. Resum

El diagrama de l'arquitectura és pot simplificar tal com es veu a la Fig 6.1.

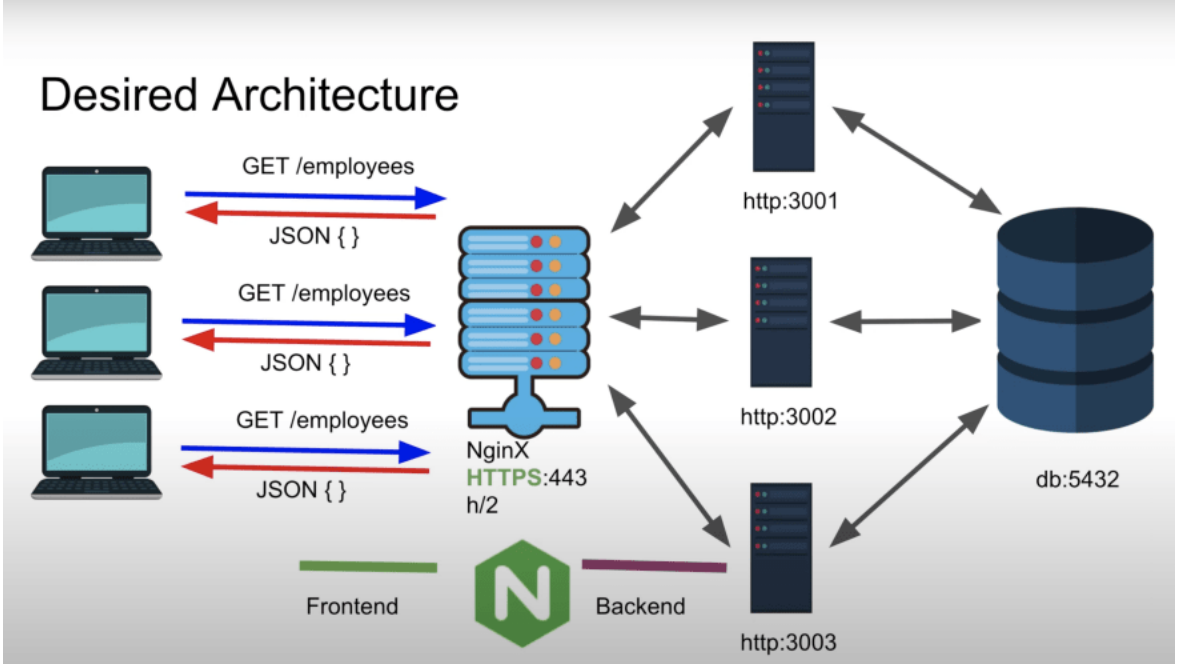


Fig 6.1. Esquema de l'arquitectura del projecte. Font: Aemie Jariwala, 2021

En resum, un client realitza una petició, llavors el servidor web NGINX realitza una conversió binària actuant com a *reverse Proxy* utilitzant el protocol uswgi cap al servidor uWSGI, que pot comunicar-se amb els *scripts* de Python. USWGI demana a Flask si coneix la URL que li ha arribat, i Flask li retorna la app en Dash corresponent (o missatge d'error 404 en cas de no trobar-ne) o modifica les dades de la mateixa (comunicant-se amb les BD o no) depenent del tipus de petició.

6.1.4. Estructura de fitxers i codi

Per al servidor del WDO, el camí per al projecte és ***/srv/wcdo.***

Després, es presenten les següents carpetes:

- ***/src_data*** per als *scripts* per extraure i processar dades.

- ***/src_viz/*** per als *scripts* i apps per a visualitzar.
- ***/databases/*** per a les BD tant les que tenen *_production* (utilitzades al web) com les que no (sota creació).
- ***/datasets/*** per tenir la versió comprimida en.CSV d'algunes BD.
- ***/dumps/*** per alguns dumps de Wikimedia, tot i que els *scripts* solen llegir-los directament des del servidor, ja que estan sota la mateixa xarxa.
- ***Venv/*** per l'entorn virtual que s'utilitza per instal·lar els mòduls de Python.
- ***Other/*** per a miscel·lània de versions antigues (*deprecated*).

6.1.5. Temàtiques del WDO

El WDO se centra en la diversitat i en els forats de les categories explicades a l'apartat 2.2. Principalment se centra en el CCC, el contingut de context cultural, que representa el contingut només present en un context cultural concret, per exemple un plat molt típic d'una regió, un ball tradicional, costums, religions, etc. Les visualitzacions i eines permeten fer-se una idea molt bona de l'estat d'aquest tipus de forat de contingut, doncs disposa de solucions amb molts tipus de gràfics i representacions diferents per conèixer aproximacions de gènere, geogràfics, etc. A més de llistes per cercar CCC de tot tipus utilitzant paraules clau.

Per entendre cada eina i visualització en profunditat, es recomana la visita a cadascuna de forma individual a https://wdo.wmcloud.org/.

6.2. Especificacions tècniques

6.2.1. Decisions tecnològiques

El control de versions del projecte es duu a terme mitjançant la plataforma ***GitHub***, doncs el projectista té experiència amb aquesta.

6.2.1.1. Llenguatge de programació

Per a l'elecció de l'entorn de desenvolupament s'utilitzen criteris usuals a qualsevol projecte de software, afegint els d'un desenvolupament Open Source:

- **Usabilitat** i experiència prèvia del projectista.
- **Accessibilitat,** ha de ser completament gratuït i accessible per complir amb l'estàndard Open Source.

- **Adequació**, normalment els llenguatges estrictament orientats a objectes no són adients per l'anàlisi i el tractament de dades.
- **Comunitat i Documentació**, es creu vital tant per resoldre dubtes durant el desenvolupament com per a que el software tingui continuïtat i manteniment.
- **Llibreries** que facilitin la tasca a realitzar i que compleixin els anteriors criteris.
- **Compatibilitat** amb el punt de partida del projecte (WDO).

Tenint en compte els criteris esmenats anteriorment, de tres alternatives proposades llistades i puntuades atenent als criteris i puntuacions de la Taula 6.1, s'ha escollit només una que permeti tant desenvolupament web com de programari per simplificar el projecte i les necessitats d'informació.

Entorn	**Usabilitat**	**Accessibilitat**	**Adequació**	**Comunitat i Documentació**	**Llibreries**	**Compatibilitat**	**Total**
Python	9	SI	10	10	10	10	49
R	6	SI	8	7	9	5	35
Javascript	8	SI	10	9	9	7	43

Taula 6.1. Puntuació de l'anàlisi d'alternatives dels 3 llenguatges de programació escollits.
Font: Elaboració pròpia

Finalment, s'utilitzarà Python com a llenguatge de programació per les següents raons:

- **Usabilitat**: És un llenguatge molt intuïtiu, que permet dur a terme accions complexes en molt poques línies de codi, fet que facilita la lectura i el manteniment. A més, té una estructura pseudo-orientada a objectes, que facilita la estabilitat i modularitat amb la qual el projectista té més experiència.

- **Accessibilitat**: És completament gratuït i compta amb IDEs molt potents com PyCharm de Jetbrains. Té una instal·lació molt senzilla i lleugera, perfecta per usuaris novells.

- **Adequació**: Python no només s'utilitza en *data science*, però no treu que el llenguatge s'utilitzi principalment per anàlisi de dades i desenvolupament web [20], que és el que es duu a terme al present projecte. L'adequació també ve facilitada per les llibreries.

- **Comunitat i documentació:** Tot i no tenir una documentació tan específica i ben ordenada com la de Java, té una comunitat gegant, i és que Python és el llenguatge més popular segons l'índex PYPL [21], basat en la quantitat de cops que es busquen tutorials a Google.

- **Llibreries:** És el criteri de més pes, doncs compta amb llibreries com NumPy, Pandas i SciPy pel tractament i anàlisi de dades i Plotly i Matplotlib per fer visualitzacions molt potents amb poquíssimes línies de codi. A més, compta amb Dash per crear aplicacions web sense haver de codificar HTML ni CSS, punt feble del projectista. Finalment, també compta amb llibreries per interactuar amb la Viquipèdia i els seus projectes com Pywikibot[18] per manegar la Viquipèdia i fer edicions etc. i Toolforge[19], per connectar-se en dues línies a les BD rèpliques.

- **Compatibilitat:** El WDO està desenvolupat en Python, així que hi ha una molt bona compatibilitat en integrar el resultat del projecte al WDO.

R té menys llibreries, és més complicat usar-lo al web i és menys usable, tot i que és molt potent i s'utilitza àmpliament en *data science*. Precisament per aquesta potència, no és tan adient, ja que el projecte no requereix de visualitzacions potentíssimes i patint el cost d'oportunitat.

Javascript no s'ha escollit per tenir llibreries amb menys documentació i no tan usades com les de Python, a més de manca d'experiència del projectista i de compatibilitat amb el WDO.

IDE

Per a l'entorn de desenvolupament, s'ha optat per *PyCharm Professional*, ja que el software de *Jetbrains*, malgrat el seu consum de RAM, és un programa molt potent i el projectista gaudeix de clau educativa per poder usar-lo, a més de certa experiència en el seu ús. És un IDE usable, amb gran quantitat d'ajudes per a l'escriptura de codi com suggeriment intel·ligent, corrector ortogràfic integrat, compatibilitat i tipificació de sentències SQL, etc.

[18] https://www.mediawiki.org/wiki/Manual:Pywikibot

[19] https://wikitech.wikimedia.org/wiki/User:Legoktm/toolforge_library

Llibreries

Principalment s'utilitzen les llibreries externes Dash, Plotly Express i Pandas.

- **Plotly** és una llibreria gràfica interactiva, open-source i basada en navegadors per a Python, sobre la qual es construeixen noves llibreries.

    - **Express**[20] és un mòdul de Plotly que conté funcions per crear figures completes a la vegada, en una sola instrucció. Totes les figures creades mitjançant express, es poden crear mitjançant les figures de Plotly, però invertint entre cinc i cent vegades més línies de codi. És particularment rellevant pel projecte ja que és una llibreria de molt alt nivell, que permet crear figures i gràfics molt atractius visualment en molt poques línies de codi, amb bona documentació darrera i una gran comunitat per resoldre dubtes.

    - **Dash**[21] és el *framework* de Python més descarregat per a la creació de web apps per *Machine Learnin*g i *data science*. Construït sobre Plotly.js, React i Flash. Dash converteix elements d'interfície d'usuari moderna com *dropdowns*, *sliders* i gràfics directament a codi Python nadiu. És especialment rellevant pel projecte ja que permet fer desenvolupament *full-stack* amb un sol llenguatge de forma senzilla. A més, té una compatibilitat extraordinària amb Dash, doncs ambdues s'emmarquen en el projecte Plotly, fet que permet desplegar webs molt vistosos de forma ràpida, senzilla i elegant. Per tal de fer els desplegaments, Dash utilitza Flask, un micro-*framework* que permet muntar aplicacions sobre servidors uWSGI.

- **Pandas**[22] és una ena ràpida, potent, flexible i fàcil d'utilitzar i Open-Source per anàlisi i manipulació de dades, construïda sobre Python. És molt rellevant pel projecte ja que permet manegar conjunts de dades de forma molt eficient i pràctica, ordenant, classificant, separant, ajuntant i un llarg etc. en molt poques línies de codi ,que també s'integra molt be amb les llibreries de Plotly. Fins a tal punt que en diversos exemples

[20] https://plotly.com/Python/plotly-express/

[21] https://plotly.com/dash/

[22] https://pandas.pydata.org/

de la documentació de les llibreries esmenades anteriorment, s'utilitza Pandas per tractar els datasets.

6.2.1.2. Entorn de Base de Dades

Per a l'emmagatzematge i l'extracció de dades s'utilitzarà SQLite3[23], compatible amb Python i fàcil d'utilitzar. És el motor de base de dades més desplegat al món, amb retrocompatibilitat i compromesos a mantenir el servei almenys fins a 2050. Es gratuït, fàcilment accessible i també s'utilitza al WDO.

6.2.1.3. Desplegament

Pel *hosting* del web s'utilitzarà un servei gratuït de Wikimedia Cloud Services[24], ja sigui Cloud VPS o Toolforge que proveeixen serveis gratuïts per als projectes vinculats amb el moviment i la creació d'eines, tenen bon ample de banda i s'utilitza en les eines del moviment, per tant no té sentit usar-ne d'altres. A més, en tractar-se d'una extensió del WDO, que efectivament s'allotja a WM Cloud Services, el projecte seguirà els mateixos passos.

6.2.2. Estructura i modelat de dades

Abans de poder treballar amb l'estructura de MediaWiki de forma directa, cal tenir accés a aquesta, mitjançant la creació d'un compte de Wikitech, crear un compte de desenvolupador i demanar accés a l'espai d'eines Toolforge.

Així i tot, es poden usar diverses eines abans de tenir-hi accés per poder interactuar amb les fonts de dades mitjançant Python, com per exemple fer peticions REST a les API de MediaWiki. A més, es pot gaudir de l'eina Quarry, el Wikidata Query Service (WDQS) i la *sandbox* de l'API de Wikipedia per fer proves i familiaritzar-se amb l'esquema, l'accés i l'extracció d'informació de les BD.

---

[23] https://www.sqlite.org/mostdeployed.html

[24] https://en.wikipedia.org/wiki/Wikipedia:Wikimedia_Cloud_Services

L'estructura de BD de qualsevol projecte MediaWiki es pot consultar al portal "*Manual:*" de la mateixa[25].

Pel que fa a aquest projecte, principalment es treballa amb dades de la BD del WDO ***wikipedia_diversity.db***, ja que conté moltes dades sobre diversitat per a tots els articles de totes les edicions lingüístiques de Viquipèdia que s'actualitza mensualment, amb més de 100 columnes per emmagatzemar dades rellevants A més, aquestes dades es complementen fent crides a les BD de Viquipèdia de cada llengua o a Wikidata per contrastar dades en temps real.

Addicionalment, es crea la BD ***gender_homepage_visibility.db*** per emmagatzemar les ocurrències de cada gènere de les persones amb biografia citades a les pàgines principals de totes les edicions lingüístiques. La BD consta només d'una taula, ja que s'emmagatzemen les dades en *long format*, també anomenat *tidy*, de manera que aquest tipus de dades conté una fila per observació, i una columna per variable tal com es veu a la Fig 6.2.

```
CREATE TABLE persons (
    lang      VARCHAR   NOT NULL,
    timestamp TIMESTAMP NOT NULL,
    gender    INTEGER   NOT NULL,
    person    INTEGER   NOT NULL,
    PRIMARY KEY (
        lang,
        timestamp,
        gender,
        person
    )
);
```

Fig 6.2. Sentencia DDL de creació de la taula *persons* de la BD ***gender_homepage_visibility.db.*** Font: Elaboració pròpia, 2021

Aquesta BD permetrà el desenvolupament de la solució explicada al subapartat 6.3.2.

[25] https://www.mediawiki.org/wiki/Manual:Database_layout

6.2.3. Enginyeria del Software

Per al disseny i l'estructura del codi s'ha decidit optar per una aproximació semblant a la del WDO. Així doncs, s'escull l'estructura MVC (Model, Vista, Controlador) per al codi propi, tot i que sense interferir en el codi ja desenvolupat del WDO, aprofitant el que ja s'ha realitzat de forma correcta. En aquest cas, el Vista-Controlador es pot combinar, ja que Dash incorpora *@Callbacks*, que permeten interactuar amb els elements de la pàgina de forma senzilla. Doncs, el model es pot simplificar tal com es veu a la Fig 6.3.

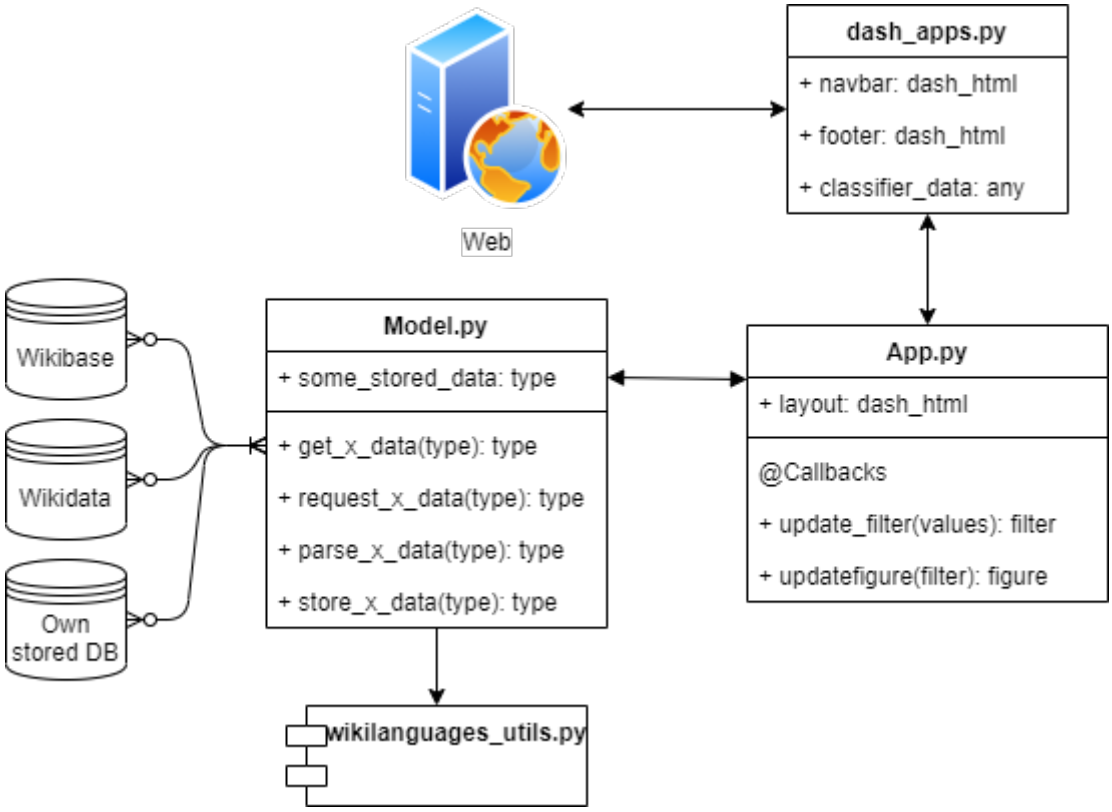


Fig 6.3. Model UML simplificat del codi del projecte. Font: Elaboració pròpia, 2021.

La classe que actua com a Model no té cap coneixement del que passa a la Vista-Controlador (App.py), només recull i classifica les dades, i les emmagatzema en una DB o document si s'escau, podent utilitzar la classe wikilanguages_utils.py que actua com a llibreria de mètodes comuns en diferents classes per al WDO. Aquestes dades poden provenir de Wikidata, utilitzant consultes SPARQL, de Wikibase, on es troben totes les rèpliques de les bases de dades dels projectes germans (enwiki,cawiktionary,etc.) o bé d'una DB d'elaboració pròpia, depenent de l'escenari.

Quan arriba una petició al web, la classe dash_apps.py redirecciona el client a una de les Apps. Aquesta també depèn de dash_apps.py, doncs els elements HTML per a la barra de navegació i el peu de pàgina són iguals i estan emmagatzemats en aquesta.

L'App.py mostra les seves dades amb els seus filtres utilitzant Dash i Plotly Express. Quan l'usuari interactua amb aquestes filtres, es genera un *Callback*, que es recull en el mateix *script*, i si les dades reflectides a la visualització s'han d'actualitzar, demana les dades al Model.py o si una és una operació senzilla, al mateix *script*, passant per paràmetre els filtres, i construeix la figura amb les noves dades.

A les apps Medical Articles, Monuments and Buildings Articles i Map of Geolocated Articles, modificar un valor als filtres, fa que es cridi un *callback* que codifica el valor dels paràmetres dins la URL. Quan es refresca la pàgina, un altre *callback* descodifica la URL i passa els paràmetres a la funció que construeix la pàgina sencera.

6.2.4. Desplegament i posada en marxa

Pel que fa a la posada en marxa, es poden trobar dos escenaris, un de particular i l'altre de comú. El particular es dona degut a la necessitat de tenir un *script* que ha d'obtenir recurrentment dades, concretament el que omple la BD ***gender_homepage_visibility.db***. Per tant, primer es tractarà el cas particular i després el comú: la implementació de l'APP web.

6.2.4.1. Desplegament d'*scripts* d'obtenció recurrent de dades

Per realitzar el desplegament, primer s'ha de programar l'execució de l'*script*. En aquest cas, el ***gender_homepage_visibility_metrics.py*** perquè realitzi l'obtenció de dades de forma repetitiva en un interval determinat de temps. Es pot fer mitjançant el sistema operatiu del servidor o bé fent un bucle infinit des de Python que l'executi esperant 86.400 segons entre execució i execució (un dia sencer) amb la funció ***sleep(seconds)*** del mòdul original de Python, ***time***. S'ha optat per la segona opció per recomanació del tutor per no interferir amb altres *scripts* en producció.

Ara, cal demanar al servidor que executi aquest *script* (que continuarà corrent fins que algú el pari o hi hagi un error). Per tant, per tal d'executar el programa dins del servidor, es crea l'*script* en ***shell*** amb extensió.***sh*** com es veu a la Fig 6.4.

```
#!/bin/bash

cd /srv/wcdo/
source venv/bin/activate
cd /srv/wcdo/src_viz/apps_dev
python3 -u gender_homepage_visibility_metrics.py > gender_homepage_visibility_metrics.log
```

Fig 6.4. Script en shell per a l'execució del programa gender_homepage_visibility_metrics.py. Font: Elaboració pròpia, 2021

Aquest *script* primer obre el terminal i navega fins el directori correcte utilitzant la comanda **cd**, després activa l'entorn virtual de Python i navega cap al directori del programa, l'executa i crea un arxiu de *logging* per recollir els missatges sortints del programa.

Ara, cal executar l'*script* ***gender_homepage_visibility_metrics.sh*** just creat utilitzant la comanda ***sudo./gender_homepage_visibility_metrics.sh & disown*** o al directori del programa. El paràmetre ***disown*** converteix el servidor (l'usuari *root*) en amo del procés, per tal de que s'executi al servidor i no a la màquina de qui envia la comanda.

En cas d'afegir algun altre programa d'aquestes característiques, només cal intercanviar el nom de l'exemple pel desitjat i afegir el bucle infinit dins del codi de Python del programa o bé programar la seva execució des de ***bash*** amb ***crontab***[26]**.**

6.2.4.2. Desplegament d'APP web

Primerament, cal actualitzar l'*script* ***dash_apps*** (***dash_apps_dev*** en aquest cas), que s'encarrega del servidor web. Amb l'extensió que s'ha assignat a l'APP, per exemple ***/gender_homepage_gap***, es crea un component HTML per afegir la nova APP a la barra de navegació com es mostra a la Fig 6.5.

[26] https://web.archive.org/web/20151226000333/http://www.debian-tutorials.com/crontab-tutorial-cron-howto

```
navbar = html.Div([
    html.Br(),
    dbc.Navbar(
        [ dbc.Collapse(
                dbc.Nav(
                    [
                    dbc.DropdownMenu(
                        [dbc.DropdownMenuItem("Monuments and Buildings Articles",
                                              href="https://wdo-dev.wmcloud.org/monuments_and_buildings_articles/"),
                        dbc.DropdownMenuItem("Medical Articles", href="https://wdo-dev.wmcloud.org/medical_articles/"),
                        ],
                        label="Tools",
                        nav=True,
                    ),
                    dbc.DropdownMenu(
                        [dbc.DropdownMenuItem("Gender Home Page Gap",href="https://wdo-dev.wmcloud.org/gender_homepage_gap/"),
                        dbc.DropdownMenuItem("Map of Geolocated Articles", href="https://wdo-dev.wmcloud.org/map_of_gaps/"),
                        ],
                        label="Visualizations",
                        nav=True,
                    ),
                    html.A(
                    # Use row and col to control vertical alignment of logo / brand
                    dbc.Row(
                        [...],
                        align="center",
                        no_gutters=True,
                    ),
                    href="https://meta.wikimedia.org/wiki/Wikipedia_Diversity_Observatory", target= "_blank",
                style = {'margin-left':"5px"}),
                ], className="ml-auto", navbar=True),
                id="navbar-collapse2",
                navbar=True,
            ),
        ],
        color="white",
        dark=False,
        className="ml-2",
    ),
    ])
```

Fig 6.5. Creació i assignació de la barra de navegació 'navbar' del web https://wdo-dev.wmcloud.org/. Font: Elaboració pròpia, 2021

També cal importar l'APP a **dash_apps_dev.py** perquè es pugui compilar i executar correctament: ***from apps_dev.gender_homepage_app import** ** (de l'*script* gender_homepage_app, que es troba a la carpeta apps_dev, importa tot).

Amb el fragment que es veu a la Fig 6.6, assigna i es corre l'APP web, que l'*script* ***wsgi_dev.py*** utilitzarà per fer córrer el servidor web.

```
##### FLASK APP #####
app_dev = flask.Flask(__name__)

if __name__ == '__main__':
    app_dev.run_server(host='0.0.0.0', threaded=True)
```

Fig 6.6. Fragment de codi corresponent a l'execució de l'APP Flask. Font: WDO, 2021.

Finalment, cal utilitzar la comanda ***sudo systemctl restart dash_apps_dev*** per aturar el servidor web i tornar a compilar-lo havent importat l'*script* de l'APP, procés que triga aproximadament entre tres i cinc minuts. ***Systemctl*** és la comanda per utilitzar ***systemd***, un sistema *init* i alhora un administrador del sistema Linux. Aquesta comanda utilitza l'arxiu ***dash_apps_dev.ini*** per saber quines instruccions ha d'executar. Les instruccions que es poden veure a la Fig 6.7, en resum, fan el següent:

- Importar l'APP que ha de córrer al servidor uwsgi, que es troba a l'*script* ***wsgi_dev.py*** amb el nom "app_dev".
- Assignar fins a quatre processos al procés.
- Assignar un *socket* de comunicació per saber com comunicar client i servidor.
- Assignar uns documents de registre de peticions REST i d'errors respectivament.

```
[uwsgi]
module = wsgi_dev:app_dev

master = true
processes = 4

socket = /srv/wcdo/src_viz/dash_apps_dev.sock
chmod-socket = 660
vacuum = true

die-on-term = true

req-logger = file:/srv/wcdo/src_viz/reqlog_dev
logger = file:/srv/wcdo/src_viz/errlog_dev
```

Fig 6.7. Codi de l'arxiu.ini que utilitza el servidor web. Font: WDO, 2021.

Degut a que el desplegament de les diferents solucions per a cada grup d'interès, cauen dins del ventall d'escenaris explicat en aquest subapartat, tornar a mencionar-los seria redundant i no s'explicaran a l'apartat 6.3.

6.3. Producte

En aquest apartat, es redacta detalladament el desenvolupament de cadascun dels productes per a cada grup d'interès, seguint els passos explicats a la metodologia.

Primerament, cal destacar la pendent de la corba d'aprenentatge que ha suposat el descobrir entorns nous com el manegament de dades de la Wikipedia i Wikidata per al desenvolupament, fet que ha endarrerit les fites previstes pel projecte, doncs ha calgut investigar a fons la millor manera de realitzar les aproximacions a cada visualització que han patit modificacions a partir de les iteracions sobre el producte.

6.3.1. Grups d'interès

Els grups d'interès escollits com a *stakeholders* i *target* del projecte i que es desenvolupen a continuació són els següents:

- Col·lectius de forat gènere[27]
- Wiki Project Med (Wikimedia Medicine)[28]
- Wikimaps User Group[29].
- WikiClassics User Group[30] i Wiki World Heritage User Group[31].

6.3.1.1. Visibilitat de gènere - Col·lectius de forat gènere

Els grups escollits per al primer producte són tots aquells que comparteixen l'objectiu de visibilitzar o acabar amb el forat de gènere que hi ha a la Viquipèdia.

La raó no necessita una profunda explicació, si es vol assolir la suma del coneixement humà, no es pot desacreditar o deixar de banda a més de la meitat de la població humana, per això cal donar a conèixer aquesta manca de pluralitat de gènere a la pàgina més vista de la Viquipèdia.

6.3.1.2. Articles geolocalitzats - Wikimaps User Group

Aquest grup d'usuaris busca agrupar usuaris que treballin amb quelcom que tingui relació amb mapes, dades geogràfiques o afegir coordinades als articles de la Viquipèdia. Degut a la necessitat d'actualitzar el paradigma de la geografia aprofitant el salt tecnològic de la

[27] https://meta.wikimedia.org/wiki/Gender_gap/Groups

[28] https://meta.wikimedia.org/wiki/Wiki_Project_Med

[29] https://meta.wikimedia.org/wiki/Wikimaps_User_Group
[30] https://meta.wikimedia.org/wiki/WikiClassics_User_Group
[31] https://meta.wikimedia.org/wiki/Wiki_World_Heritage_User_Group

última dècada [22] i la necessitat de fer recerca en geografia per a l'educació [23], es decideix ajudar a aquest grup d'usuaris a omplir forats de contingut referent a articles geolocalitzats (articles amb coordenades) en format Latitud, Longitud.

6.3.1.3. Articles de medicina - Wiki Project Med (Wikimedia Medicine)

La medicina és un camp que permet millora la vida dels que viuen a la Terra, fet que motiva al projectista a contribuir a la causa, encara més tenint en compte l'impacte que ha tingut la pandèmia de la COVID-19 a la vida a la que la humanitat estava acostumada. A més, la Viquipèdia és realment rellevant en l'aspecte mèdic, gràcies a estar demostrat que els practicants utilitzen extensivament la Viquipèdia per donar tractament [24] i es considera que "malgrat alguns dels articles més populars de tots [...] tenen una molt bona qualitat o estan destacats, d'altres, també populars, necessiten millora." [24]. És per això que es decideix ajudar aquest col·lectiu, a més d'ajudar a academitzar la Viquipèdia, millorant els articles perquè els acadèmics més crítics confiïn en la Viquipèdia com a enciclopèdia de referència.

6.3.1.4. Articles d'edificis i monuments WikiClassics User Group i Wiki World Heritage User Group

El grup WikiClassics es centra en la cultura clàssica, primerament però no única, en l'antiga Grècia i l'antiga Roma dels projectes Wikimedia. El seu objectiu principal és millorar la qualitat i la quantitat d'informació sobre l'antiguitat clàssica a les plataformes Wikimedia.

El grup WIki World Heritage busca cobrir tot els emplaçaments Patrimoni de la Humanitat de la UNESCO[32] arreu del món. Actualment, hi ha 1121 localitzacions a 167 estats, i mentre alguns ja tenen afiliats amb els que busquen col·laborar, molts altres estats no en tenen i busquen arribar a les comunitats locals i assegurar-se que "ningú es queda enrere". És a dir, que busquen omplir forats de CCC.

Aquesta és una proposta interwiki, doncs la seva utilització podria ser positiva per a dos projectes germans, la Viquipèdia i Viquiviatges, la guia turística lliure basada en la tecnologia wiki. Millorant o omplint forats el contingut sobre edificis i monuments, facilitaria el crear guies més completes en més idiomes en disposar de més informació per a

[32] https://whc.unesco.org/en/about/

la seva elaboració. A més, arran de les restriccions de mobilitat causades per la pandèmia de COVID-19, podria ajudar a potenciar el turisme local i el CCC. És per aquestes raons que es decideix ajudar els col·lectius de cultura clàssica i de patrimoni mundial en la seva labor, doncs a finals de juny del 2020, dels més de mil emplaçaments amb el reconeixement de patrimoni de la humanitat, el 44% no tenen un article en anglès, 45 en castellà i s'accentua fins al 86% en àrab [25].

6.3.2. Visualització: Homepage Gender Visibility

Aquesta solució permet obtenir una idea de la varietat de gènere de les persones que se citen a la pàgina principal o portada de les diferents Viquipèdias mitjançant un gràfic de barres amb percentatges i nombre d'ocurrències. A més, permet seleccionar un interval de dies per comptar. El resultat es pot veure a https://wdo-dev.wmcloud.org/gender_homepage_gap/

6.3.2.1. Disseny de datasets i *dashboards*

L'objectiu d'aquest producte és donar visibilitat. Per tant, s'escull com a punt de partida la pàgina més visitada de la Viquipèdia: la Portada [26]. Doncs, es vol computar el gènere dels *outlinks* (enllaços a altres articles de la Viquipèdia) a biografies que apareixen a la Portada de cada edició lingüística.

Llavors, és necessari seguir un procediment per obtenir la tupla resultant que es pot veure a la Fig 6.8.:

Person
(lang,timestamp,gender,person)

Pablo Picasso
(ca,1618952730,Q6581097,Q5593)

Fig 6.8. Exemple de tupla resultant on es recull la llengua origen, la marca de temps, l'identificador de gènere i l'identificador de la persona a Wikidata respectivament. Font: Elaboració pròpia, 2021.

Els passos són:

- Obtenir el ***page_id*** de la Portada de cada edició lingüística.
- Per cada ***page_id***, obtenir els ***pagelinks***

- Per a cada ***pagelink,*** obtenir el seu identificador de Wikidata, és a dir el seu ***wikibase_item*** (Q5593 per a Picasso, per exemple).
- Donat un identificador, comprovar que sigui una instància (P31) d'humà (Q5) i comprovar que tingui la propietat sexe o gènere (P21). Si això es compleix, guardar el valor del seu identificador (Q5593), el seu sexe o gènere (Q6581097), l'edició lingüística on apareix i el ***timestamp*** [33] o marca de temps en la que s'ha realitzat la cerca, per poder tenir una dimensió de temps.

D'aquesta manera, guardant els identificadors en comptes dels noms, no hi ha confusió possible en la traducció o interpretació del gènere o sexe o de la persona a qui fa referència. A més, permet filtrar per edició lingüística, per gènere, per persona (ja que pot aparèixer la mateixa persona en diferents Viquipèdias) i per temps.

Aquesta tupla es guarda en una BD local per poder tenir la variable de temps. D'altra banda, si es vol prescindir d'aquesta, les consultes es poden realitzar en temps real per a la Portada del dia en que es faci la cerca, doncs aquesta canvia diàriament.

6.3.2.2. Desenvolupament

En la primera iteració, es va optar per utilitzar ***pywikibot***, una llibreria per interactuar amb Wikimedia des de Python, iterant sobre totes les pàgines de Portada, però en veure que amb una sola Portada, trigava al voltant d'entre tres i cinc minuts, es va desestimar, ja que per a 300 edicions lingüístiques, això equivaldria a vint-i-cinc hores de processament. Doncs, s'opta per un altre aproximació arrel dels suggeriments del tutor: utilitzar les crides a les BD per optimitzar la cerca.

Després de provar tres aproximacions diferents, consultar a pàgines de Wikidata amb altres usuaris i conversar per IRC amb d'altres, es va aconseguir l'aproximació definitiva, combinant els avantatges de l'API de MediaWiki i les consultes al servei de consultes de Wikidata (WDQS) mitjançant el llenguatge SPARQL, que també ha suposat una corba d'aprenentatge considerable.

Passos a seguir

[33] Marca de temps Unix. Compta els segons que han passat des del 01/01/1970 (UTC)

Per començar, cal aconseguir els ***page_id*** de les Portades. Afortunadament, només cal aconseguir-les un cop, ja que és un identificador que no varia. Per tal d'aconseguir-les, s'han escrit dues funcions: ***getMainPageTitles()*** i ***getPageIDByName(langcode,mainPagename)*** que es poden veure a la Fig 6.9.

```
def getPageIDByName(langcode:str,mainPageName: str):

    try:
        wikipedia.set_lang(langcode)
        page = wikipedia.page(title=mainPageName)
        id = page.pageid
        return id

    except Exception as e:
        print(e)
        print(f'Something wrong with {langcode} and {mainPageName}')

def getMainPageTitles():
    site = pywikibot.Site('wikidata', 'wikidata')
    repo = site.data_repository()
    item = pywikibot.ItemPage(repo, "Q5296")
    sitelinks = item.iterlinks(family='wikipedia')

    lang_dict = {}
    for link in sitelinks:
        val = str(link)
        val = val.replace('[', '')
        val = val.replace(']', '')
        val = val.split(':')

        element = {val[0]: val[1]}

        lang_dict.update(element)

    print(lang_dict)
```

Fig 6.9. Funcions per obtenir els page_id de cada portada en cada Llengua. Font: Elaboració pròpia, 2021.

Amb la ***getMainPageTitles()*** s'utilitza ***pywikibot*** per extreure una per una els noms de les portades de cada Viquipèdia des de la pàgina de Wikidata per a les Pàgines Principals de MediaWiki, amb l'identificador **Q5296**. Un cop es té una col·lecció amb aquestes dades, només cal iterar sobre elles utilitzant ***getPageIDByName***, proporcionant el codi de llengua

i el nom de la Portada en aquesta llengua. Per tal de fer-ho, s'utilitza la llibreria ***wikipedia*** per a Python. Aquests resultats es guarden en un arxiu JSON per a posterior utilització.

A mesura que es feien iteracions i proves, es va trobar que alguns noms eren ambigus, doncs hi havia diferents articles amb aquest nom, així que donat el codi de llengua amb errors, s'ha cercat manualment el seu nom i el seu ***page_id*** utilitzant *Quarry*.

En iteracions posteriors, s'ha vist que aquest procés es podria haver realitzat utilitzant l'API de MediaWiki, però com només cal realitzar-lo un cop, no és imperatiu optimitzar la cerca.

Un cop es compta amb els ***page_id***, ja es pot començar a realitzar la cerca.

Es planteja una cerca iterativa, per cada edició lingüística, en el que es duu a terme el següent:

- Obtenir els identificadors de Wikidata dels *outlinks* d'una pàgina, donat una llengua i el ***page_id*** de la seva Portada utilitzant una crida a l'API de Wikipedia com es veu a la Fig 6.10.

```
def get_wikibase_items(langcode: str, main_page_id: int):
    url = f"https://{langcode}.wikipedia.org/w/api.php?action=query&format=json&prop=pageprops" \
          f"&pageids={main_page_id}&generator=links&utf8=1&gplnamespace=0&gpllimit=max"
    r = requests.get(url)
    result = parse_wikibase_response(r.json())
    return result
```

Fig 6.10. Funció per obtenir tots els identificadors de Wikidata d'una pàgina d'una edició lingüística concreta. Font: Elaboració pròpia, 2021.

- Amb aquests identificador, incloure aquests a la *query*[34] en SPARQL i realitzar una petició al Wikidata Query Service com es veu a la Fig 6.11.

```
query = "SELECT ?gender ?person WHERE { VALUES ?person { %s } ?person wdt:P31 wd:Q5; wdt:P21 ?gender. }"
newquery = query.replace('%s', queryValues)

url = 'https://query.wikidata.org/sparql'
headers = {'Content-type': 'application/sparql-query'}

r = requests.post(url, params={'format': 'json'}, data=newquery, headers=headers)
```

Fig 6.11. Query i petició REST a l'API del WDQS. Font: Elaboració pròpia, 2021.

[34] Consulta a la base de dades

- Ara cal convertir la resposta SPARQL en quelcom intel·ligible per als humans, en aquest cas en una llista de tuples amb el format de la Fig 6.8 i afegir les tuples de l'idioma actual a la llista de totes les llengües.
- Ara només cal inserir totes aquestes tuples a la base de dades creada anteriorment amb la funció de la Fig 6.12, que està pensada en format *long-form* o *tidy*, que es tradueix en una fila per ocurrència i una columna per variable.

```
def create_gender_homepage_visibility_db():
    conn = sqlite3.connect('gender_homepage_visibility_db')
    cursor = conn.cursor()
    query = "CREATE TABLE IF NOT EXISTS persons (lang varchar NOT NULL ,timestamp timestamp NOT NULL, " \
            "gender integer NOT NULL , person integer NOT NULL ,PRIMARY KEY (lang,timestamp,gender,person ))"
    cursor.execute(query)
    conn.commit()
    conn.close()
```

Fig 6.12. Funció per crear la base de dades del producte. Font: Elaboració pròpia, 2021

- Per inserir les tuples s'utilitza el fragment de codi escrit al ***main()*** per evitar les problemàtiques de la injecció SQL com es veu a la Fig 6.13.

```
def main():

    with open('langcode_mainPage_ID.json', encoding="utf8") as f:
        langcode_pageid_dict = json.load(f)

    result = get_gender_data(langcode_pageid_dict)
    conn = sqlite3.connect('gender_homepage_visibility_db')
    cursor = conn.cursor()
    query = "INSERT INTO persons (lang,timestamp ,gender,person) VALUES (?,?,?,?) ;"
    cursor.executemany(query, result)
    print('Inserted', cursor.rowcount, 'records to the table.')
    conn.commit()
    conn.close()
```

Fig 6.13. Funció ***main()*** que realitza totes les passes mencionades anteriorment. Font: Elaboració pròpia, 2011

En arribar a aquest punt, tot el procés d'adquisició de dades s'ha realitzat satisfactòriament dins la funció ***get_gender_data()***, amb els seus indicadors de temps i altres mesures per poder controlar millor el flux d'execució i s'han inserit les dades a la BD.

Un cop el model dins del marc del MVC s'ha desenvolupat, es procedeix a programar la Vista-Controlador mitjançant Dash, Pandas i Plotly Express.

La creació d'aquesta vista és molt senzilla, i l'estructura és igual que la del WDO. Per tant, només cal modificar les particularitats de cada cas. En aquest, la visualització és pràcticament idèntica a una ja existent.

L'aplicació segueix l'esquema de Dash, utilitzant els estils existents a l'*script* ***dash_apps.py*** només cal assignar un camí URL i un títol a la pàgina per crear el nucli de l'app, i després assignar-li un *layout* per poder-la visualitzar com es veu a la Fig 6.14.

```
title_addenda = ' - Wikipedia Diversity Observatory (WDO)'
title = 'Home Page Gender Visibility'

app = dash.Dash(url_base_pathname=webtype+'/homepage_gender_visibility/',
                external_stylesheets=external_stylesheets,external_scripts=external_scripts)
app.config['suppress_callback_exceptions'] = True
app.title = title+title_addenda
app.layout= html.Div([...], className="container")
```

Fig 6.14. Creació del nucli de l'aplicació Dash, important els components externs de ***dash_apps.py.*** Font: Elaboració pròpia, 2021.

Aquest *layout* consisteix d'una barra de navegació i d'un peu que també s'extrauen del codi ja creat del WDO. Per tant, només cal afegir la figura amb els seus filtres i el text explicatiu.

La figura s'actualitza mitjançant *Callbacks*, que representen una interacció de l'usuari amb l'aplicació. En aquest cas, en canviar els filtres de llengua, el callback crida una funció (***Update_barchart()***) que a la vegada crida a una funció de l'*script* del model (***get_gendercount_by_lang()***) per seleccionar les dades aplicant aquest filtre. Amb aquestes dades, primer es converteix els identificadors de gènere a llenguatge humà i després es crea i s'aplica la nova figura creada que es pot veure a la Fig 6.15.

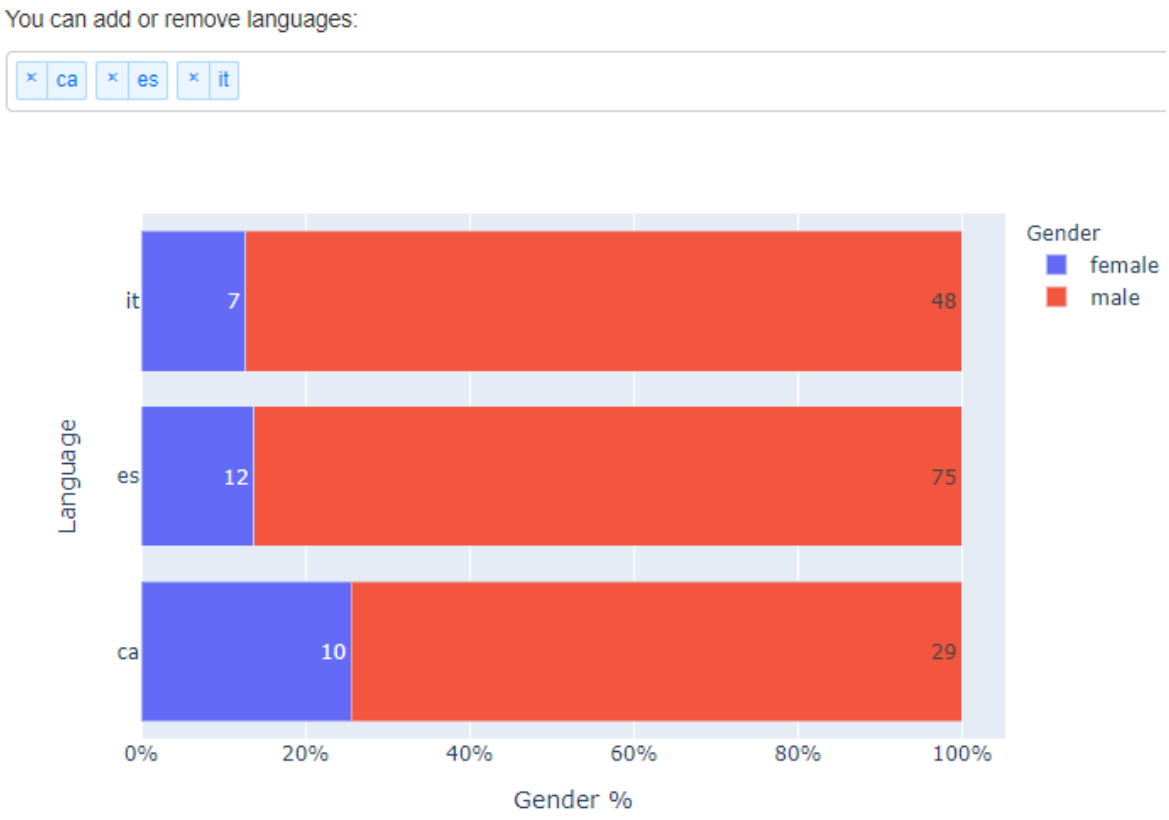


Fig 6.15. Gràfic de barres horitzontals amb els percentatges de diversitat de gènere (X) per cada llengua (Y). El color indica el gènere, blau per femení i vermell per masculí. Font: Elaboració pròpia, 2021.

Després d'haver creat i testejat la implementació al servidor, es decideix **incorporar la variable de temps** a la cerca per demostrar que la bona planificació i disseny permet afegir funcionalitats fàcilment.

Així, a la crida a la BD per actualitzar les dades, també s'inclou la variable ***timestamp*** per filtrar per dies. A més, és necessari actualitzar els filtres de cerca, incorporant un selector de dates i convertint els *timestamps* en un format interpretable pel component HTML que es pot veure a la Fig 6.16.

```
html.Div(
    html.P('Select a date range'),
    style={'display': 'inline-block', 'width': '200px'}),
dcc.DatePickerRange(
    id='date_picker_range',
    min_date_allowed=datetime.date(2021, 5, 10),
    max_date_allowed= datetime.timedelta(days=1) + datetime.datetime.utcfromtimestamp(get_last_data_update()).date(),
    start_date= datetime.datetime.utcfromtimestamp( get_last_data_update()).date(),
    end_date= datetime.timedelta(days=1) + datetime.datetime.utcfromtimestamp(get_last_data_update()).date(),
    initial_visible_month=datetime.datetime.utcfromtimestamp(get_last_data_update()).date()
), html.Div(id='output_container_date_picker_range'),
```

Fig 6.16. Element per escollir el rang de dates per filtrar la cerca. Font: Elaboració pròpia, 2021.

6.3.3. Filtres de cerca per a les eines o *tools*

En aquest subapartat es detallen les característiques dels filtres de cerca (disponibles a la Fig 6.17) que poden utilitzar les eines desenvolupades i que estan relacionats directament amb l'arquitectura de dades de ***wikipedia_diversity.db*** i el procés de creació de les APP web.

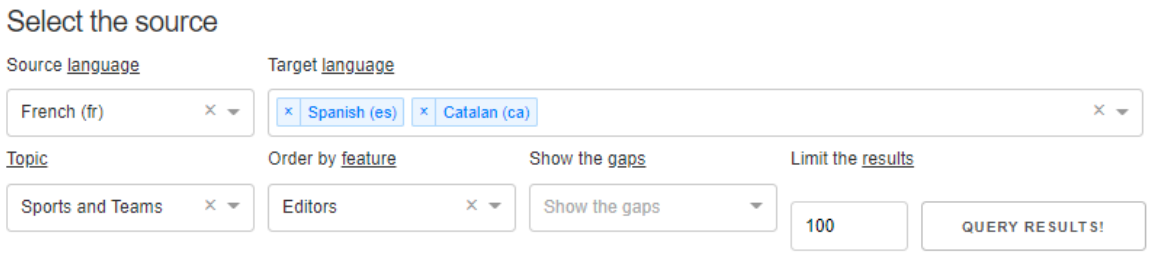


Fig 6.17. Filtres de cerca per a les eines desenvolupades. Font: WDO, 2021

Les dades necessàries per dur a terme aquesta classificació es recullen a la BD ***wikipedia_diversity.db,*** que s'obtenen mensualment mitjançant els *scripts* ***wikipedia_diversity.py***, ***content_selection.py*** i ***content_retrieval.py***. Després, es comprova la seva validesa a l'hora de mostrar la col·lecció mitjançant crides a les BD en producció de cada edició lingüística, per poder actualitzar els forats en temps real.

L'objectiu de l'ús d'aquests filtres és el de recuperar articles sobre [medicina/ monuments i edificis/ articles geolocalitzats] d'una Viquipèdia en una llengua origen i comprovar la seva disponibilitat en unes edicions lingüístiques destí específiques. A més, permet ensenyar dades sobre els articles de la llengua origen com la mida de l'article, el nombre d'editors que han participat, el nombre de visites, etc. Aquestes dades s'actualitzen mensualment. El que s'obté en temps real és la disponibilitat en les llengües destí.

6.3.3.1. Source Language

És la llengua origen, l'edició de la Viquipèdia en aquella llengua. Es pot escollir entre les més de 300 Viquipèdias existents actualment. Aquestes llengües i el seu codi estàndard s'obtenen mitjançant les funcions que ens proporciona ***wikilanguages_utils.py*** com la que es veu a la Fig 6.18.

```
def load_wiki_projects_information():
    # in case of extending the project to other WMF sister projects, it would be necessary to revise these columns and
    # create a new file where a column would specify whether it is a language edition, a wikictionary, etc.

    conn = sqlite3.connect(databases_path+diversity_categories_production_db);

    query = 'SELECT languagename, Qitem, WikimediaLanguagecode, Wikipedia, WikipedialanguagearticleEnglish, ' \
            'languageISO, languageISO3, languageISO5, nativeLabel, region, subregion, intermediateregion FROM wiki_projects;'

    languages = pd.read_sql_query(query, conn)
    languages = languages.set_index(['WikimediaLanguagecode'])


    return languages
```

Fig 6.18. Funció per obtenir informació sobre les diferents edicions lingüístiques de la Viquipèdia. Font: WDO, 2021.

Després, en una altra funció es recorre tota la col·lecció perquè les dades es vegin reflectides en el format **Llengua en anglès(codi)**. Pel cas del català seria Catalan(ca).

6.3.3.2. Target Language

És la llengua o llengües destí, en les que es vol comprovar la disponibilitat dels articles llistats de la Viquipèdia origen. La forma d'obtenir les llengües destí és igual a l'exposada al punt anterior.

6.3.3.3. Topic

Aquest filtre es tracta del tema de l'article. És el filtre que ens permet crear les col·leccions d'articles temàtiques. L'*script* que omple la BD ***wikipedia_diversity.db*** utilitza una tècnica de cerca en profunditat (DFS), recorre tots els ítems de Wikidata cercant les propietats que pertanyen a cada tema per nivells. Per exemple, la pàgina de Wikidata de la truita de patates (Q281751) és una instancia de (P31) plat (Q746549), llavors l'*script* cercaria la pàgina de plat. Plat és una instància d'objecte físic artificial (Q8205328), de producte (Q2424752) i de menjar (Q2095), que és l'identificador que fa servir l'*script* per incloure un article dins la col·lecció d'articles sobre menjar. Per representar aquesta pertinença, la BD té una columna

food, on tindrà l'identificador de plat (Q746549), que ha sigut l'identificador que ha permès arribar a l'identificador de *food* (menjar).

Així, si es vol cercar un article sobre menjar, només cal afegir la condició que la columna *food* no estigui buida. En llenguatge SQL seria: **WHERE** *food* **IS NOT NULL**

6.3.3.4. Order by feature

L'ordenació dels articles. Aquest filtre permet ordenar els articles per un indicador concret, com el nombre d'editors, la mida de la pàgina, etc. En llenguatge SQL, per ordenar per nombre de visites seria: **ORDER BY** num_pageviews.

Mitjançant un diccionari de Python, es mapeja el nom de la columna a la BD amb quelcom intel·ligible per l'usuari. En aquest cas, la dupla clau:valor seria ***{'Pageviews': 'num_pageviews'}***

6.3.3.5. Show the gaps

Aquest filtre és l'únic que cerca en temps real. Permet obtenir quatre tipus de resultats.

1- **Sense filtre**
   Recupera els articles sense tenir en compte si hi ha forat o no, ordenats pel valor d'*Order by feature*. Per tant, poden aparèixer forats o no.
2- **Almenys un forat en una llengua (At least one gap)**
   Recupera els articles que com a mínim manquen en una de les llengües destí seleccionades.
3- **Només forats en ambdues llengües (Only gaps)**
   Recupera els articles de la llengua origen que no existeixen en cap de les llengües destí.
4- **Sense forats (No gaps)**
   Recupera els articles que existeixen en totes les llengües destí.

Tal com es veu a la Fig 6.19, donat una llengua origen, llengües destí, una col·lecció de ***page_id*** de les pàgines a cercar i la connexió prèviament establerta amb les BD de Viquipèdia en producció, aquesta funció permet seleccionar els títols de les pàgines origen

en les llengües destí. En cas que no existeixin, aquell valor queda buit, que serveix posteriorment per identificar els forats.

```
# Get me the articles that are also in the wikipedia_diversity_production.db and the diversity categories it belongs to.
def get_interwikilinks_articles(sourcelang, targetlangs, df, mysql_con_read):

    page_ids = df.page_id.tolist()
    params = page_ids + targetlangs

    page_asstring = ','.join( ['%s'] * len(page_ids) )
    query = 'SELECT ll_from as page_id, CONVERT(ll_title USING utf8mb4) as page_title, ' \
            'CONVERT(ll_lang USING utf8mb4) as lang FROM langlinks WHERE ll_from IN (%s) ' % page_asstring

    page_asstring = ','.join( ['%s'] * len(targetlangs) )
    query += 'AND ll_lang IN (%s);' % page_asstring

    df_y = pd.read_sql_query(query, mysql_con_read, params = params);
    df_y = df_y.set_index('page_id')

    i = 0
    for lang in targetlangs:
        i += 1
        df_z = df_y.loc[(df_y['lang']==lang)]
        df_z = df_z.rename(columns={"page_title": "page_title_"+str(i)})
        df = df.merge(df_z["page_title_"+str(i)], how='left', on='page_id')

    return df
```

Fig 6.19. Funció de ***wikilanguages_utils.py*** que cerca les interseccions entre els articles existents a ***wikipedia_diversity.db*** i les bases de dades en producció de les edicions lingüístiques destí. Font: WDO, 2021.

6.3.3.6. Limit the results

Un simple limitador de la cerca, que permet obtenir només els primers **n** resultats, ordenats pel filtre *Order by feature*. És necessari perquè si no hi fos, la recuperació de les dades ocuparia massa temps ja que es treballa amb *Big Data*.

6.3.4. Eina: Medical Articles i Monuments and Buildings Articles

Aquestes dues solucions s'expliquen juntes, ja que són pràcticament idèntiques però canviant la temàtica, ja que, primerament, la solució Monuments and Buildings no estava planejada. En vista de la facilitat de reproducció respecte a Medical Articles, es decideix afegir aquesta solució a la col·lecció amb l'objectiu de demostrar la fàcil extensió de funcionalitats quan un codi és suficientment modular i es tenen les dades adients.

Pel que fa a la redacció, les referències a la col·lecció d'articles de medicina o la col·lecció d'articles sobre edificis i monuments o els seus noms en anglès, és intercanviable, per tant, només s'exemplifica una solució.

El resultat de Medical Articles es pot veure a:

https://wdo-dev.wmcloud.org/medical_articles/

El resultat de Monuments and Buildings Articles es pot veure a:

https://wdo-dev.wmcloud.org/monuments_and_buildings_articles/

6.3.4.1. Disseny de datasets i *dashboards*

L'aproximació a aquest producte és realitzar una llista o taula classificada de tots els articles de medicina en una llengua, utilitzant els identificadors i les propietats de Wikidata , per determinar si pertany o no al grup, i poder comparar entre edicions lingüístiques i filtrar per diverses variables com visites, mida, forats de contingut, etc.

Així, l'objectiu d'aquestes solucions és el de visualitzar articles de medicina o monuments i edificis d'una edició lingüística , incloent biografies i una gran varietat de temes, i comprovar la seva disponibilitat en unes edicions lingüístiques concretes.

6.3.4.2. Desenvolupament

Aquestes solucions han suposat les més senzilles i fàcils d'implementar, ja que, malgrat que interpretar codi aliè sense documentar suposa un repte amb una corba d'aprenentatge logarítmica, segueixen el mateix patró que l'APP ja existent "LGBT+ Articles", amb els mateixos filtres, la mateixa disposició i composició de la taula i construcció de la pàgina. Així, els aspectes que s'han hagut de desenvolupar s'expliquen a continuació, partint del codi de l'APP LGBT+ Articles.

Per començar, cal actualitzar la nomenclatura del text i de les variables que contemplen els aspectes relacionats amb LGBT per la temàtica escollida (medicina o monuments i edificis). Després cal adreçar les quèries SQL per adaptar-les també a la temàtica, aplicant el filtre **WHERE** medicine **IS NOT NULL** com es veu a la Fig 6.20.

```
query += 'r.medicine '
query += ' FROM ' + source_lang + 'wiki r '
query += 'WHERE r.medicine IS NOT NULL '

if topic != "none" and topic != "None" and topic != "all":

    if topic == 'keywords':
        query += 'AND r.keyword_title IS NOT NULL '
    elif topic == 'geolocated':
        query += 'AND (r.geocoordinates IS NOT NULL OR r.location_wd IS NOT NULL) '
    elif topic == 'lgbt_topic':
        query += 'AND r.lgbt_topic > 0 '
    elif topic == 'men': # male
        query += 'AND r.gender = "Q6581097" '
    elif topic == 'women': # female
        query += 'AND r.gender = "Q6581072" '
    elif topic == 'people':
        query += 'AND r.gender IS NOT NULL '
    elif topic == 'not_people':
        query += 'AND r.gender IS NULL '
    elif topic == 'ccc':
        query += 'AND r.ccc_binary = 1 AND percent_outlinks_to_CCC > 0.15 '
    elif topic == 'ccc_not_people':
        query += 'AND r.ccc_binary = 1 AND percent_outlinks_to_CCC > 0.15 AND r.gender IS NULL '
    else:
        query += 'AND r.'+topic+' IS NOT NULL '
```

Fig 6.20. Fragment de la construcció de la query SQL per a Medical Articles. Font: Elaboració pròpia a partir lgbt_articles_app.py, 2021

Un cop muntada la consulta a la BD, cal executar-la i muntar els resultats en un DataFrame de Pandas per poder tractar les dades amb facilitat com es veu a la Fig 6.21.

```
df = pd.read_sql_query(query, conn)
mysql_con_read = wikilanguages_utils.establish_mysql_connection_read(source_lang);
df = wikilanguages_utils.get_interwikilinks_articles(source_lang, target_langs, df, mysql_con_read)
if order_by == "none" or order_by == "None":
    df = df.sort_values(by='num_pageviews', ascending=False)
else:
    df = df.sort_values(by=order_by, ascending=False)
df = df.fillna('')
columns_dict = {'num': 'Nº', 'page_title': source_language + ' Title', 'target_langs': 'Target Langs.',
                'qitem': 'Qitem'}
columns_dict.update(features_dict_inv)
```

Fig 6.21. Execució de la query i muntatge dels resultats en un DataFrame de Pandas. Font: lgbtq_articles_app.py del WDO, 2021

Un cop s'han aconseguit les dades i estan en un format adient per treballar amb elles, es munta la taula mitjançant els components HTML de Dash iterant sobre cada fila i columna com es pot veure a la Fig 6.22.

```
elif col == 'Medicine Topic':
    label = item_labels_dict[rows['Medicine Topic']]
    df_row.append(html.A(label, href='https://www.wikidata.org/wiki/' + rows['Medicine Topic'],
                         target="_blank", style={'text-decoration': 'none'}))
```

Fig 6.22. Tractament de la columna "Medicine Topic" durant el recorregut de cada fila i columna del DataFrame. df_row és la llista amb totes les columnes de la fila. Font: Elaboració pròpia a partir de lgbtq_articles_app.py, 2021

Perquè l'usuari entengui perquè un article pertany a la col·lecció, a la columna *Medicine Topic* s'afegeix l'etiqueta de l'identificador de la llengua origen, amb un enllaç a Wikidata per obtenir més detalls. En cas de no trobar l'etiqueta en la llengua origen, s'afegeix l'etiqueta en anglès, i en cas de no haver-hi, s'afegeix a la llista com a N/A però amb l'enllaç a Wikidata igualment com es veu al mètode per omplir la llista d'etiquetes a la Fig 6.23..

```
def get_item_labels(df, langcode):
    headers = {
        'User-Agent': 'WikipediaDiversityObservatory WDO (https://meta.wikimedia.org/wiki/Wikipedia_Diversity_Observatory;'
                      'tools.wcdo@tools.wmflabs.org) python-requests/2.18.4'}
    # Divide list of items into size 50 (Limit of API)
    divided = divide_chunks(df.values.tolist(), 50)
    labels_dict = {}
    for chunk in divided:
        query = 'https://www.wikidata.org/w/api.php?action=wbgetentities&props=labels&ids={}&languages={}&format=json'.format(
            '|'.join(chunk), langcode + '|en')
        response = requests.get(query, headers=headers)
        data = response.json()
        for q in data['entities']:
            try:
                labels_dict[q] = data['entities'][q]['labels'][langcode]['value'].capitalize()
            except KeyError as e:
                try:
                    labels_dict[q] = data['entities'][q]['labels']['en'][
                        'value'].capitalize()+' (en)'  # If doesn't exist in source lang, get it in EN
                except Exception as exep:
                    print('{}: Something wrong with QITEM {} with data {}'.format(ex, q, data['entities'][q]))
                    labels_dict[q] = 'N/A' #If it doesn't have laben in either target or english, assign random
            except Exception as ex:
                labels_dict[q] = 'N/A'
                print('{}: Something wrong with QITEM {} with data {}'.format(ex, q, data['entities'][q]))

    return labels_dict
```

Fig 6.23. Funció per obtenir les etiquetes donats uns identificadors de Wikidata i una llengua origen. Font: Elaboració pròpia, 2021

Aquesta funció utilitza el mètode ***divide_chunks*** per partir la llista d'identificadors en fragments de màxim 50 elements, el màxim permès per crida a l'API de Wikimedia, ja que s'ha prioritzat la senzillesa de la crida a l'API abans que utilitzar el WQS en el llenguatge SPARQL, amb el qual el projectista té menys experiència.

Un cop es disposa de la taula en format de components HTML, es procedeix a la implementació i visualització al web mitjançant els components de Layout de Dash, per obtenir el resultat que es pot veure a la Fig 6.24.

Medical articles retrieved from Catalan Wikipedia and its coverage by the target languages

N°	Qitem	Catalan Title	Editors	Edits	Pageviews	Interwiki	Bytes	Creation Date	Medicine Topic	Target Langs.	Spanish Title
2	Q87720284	Epidèmia de febre groga de 1821	4	76	4	0	23.0k	2020-03-14	Epidèmia		
3	Q88178141	Brot de coronavirus a la Conca d'Òdena el 2020	5	49	4	0	7.2k	2020-03-26	Brot epidèmic		
4	Q35027881	Maria Teresa Estrach i Panella	3	19	1	0	4.6k	2019-02-06	Dermatologia		
5	Q59466726	Epidèmia de febre groga de Barcelona de 1870	3	16	1	3	3.5k	2020-03-14	Epidèmia	es	Epidemia de fiebre amarilla de Barcelona de 1870

Fig 6.24. Exemple de taula de Medicine Articles. Font: https://wdo-dev.wmcloud.org/medical_articles/?target_langs=es%2Cfr&topic=ccc&source_lang=ca&show_gaps=one-gap-min&limit=100&order_by=none, 2021

6.3.5. Visualització i Eina: Map of Geolocated Articles

Aquesta solució mixta permet veure al mapa els articles que s'enumeren en una taula com la de les solucions anteriors, afegint també a la taula les coordenades de l'article, amb un enllaç per veure la ubicació a *Google Maps*. El mapa de la solució permet filtrar els articles per la seva disponibilitat en les llengües destí interactuant amb la llegenda del mateix. El resultat es pot veure a: https://wdo-dev.wmcloud.org/map_of_gaps/.

6.3.5.1. Disseny de datasets i *dashboards*

Es vol presentar una solució que permeti interpretar un mapa que mostri les localitzacions de cada article amb unes senzilles dades identificatives , concretament, les coordenades, el nom de l'article, l'identificador de Wikidata i les llengües en les que l'article està disponible. A més, es vol aprofitar tot el potencial de les eines ja desenvolupades i incloure la llista dels articles per poder disposar de més dades, incloent la columna *Geocoordinates* amb un enllaç per poder veure en un mapa extern més detallat la localització.

Per tal de desenvolupar aquestes funcionalitats, es decideix utilitzar un *Bubble Map* de Dash, de manera que cada article representi una bombolla al mapa i la seva mida estigui determinada per la variable *Order by feature* escollida per l'usuari, que per defecte serà el nombre de visites de l'article en la llengua origen.

6.3.5.2. Desenvolupament

Primer s'opta per implementar la llista, ja que només cal seguir els mateixos passos que per a Medical Articles, eina ja desenvolupada i desplegada. Per tant, s'explicarà el principalment el desenvolupament per implementar el mapa. Per saber-ne més sobre el desenvolupament de la llista o taula, vegeu el subapartat 6.3.4.2.

Taula

Per la columna *Geocoordinates* de la taula, es barallen dues possibilitats pel que fa a l'enllaç de les coordenades: Google Maps o Open Street Maps. Open Street Map és un projecte Open Source i col·laboratiu, per tant, alineat amb l'objecte del projecte. Finalment s'opta per Google Maps, degut a la seva usabilitat i familiaritat dels usuaris. A més, permet veure les localitzacions a peu de carrer amb *Street View* i conté ressenyes i fotografies d'altres usuaris. Així i tot, es desenvolupen les dues opcions, deixant comentada la opció que no s'utilitza, per si es canvia d'opinió tal com es veu a la Fig 6.25.

```
elif col == 'Geocoordinates':
    # GoogleMaps
    df_row.append(
        html.A(rows['Geocoordinates'].replace(',',', '),
               href='https://www.google.com/maps/search/' + rows['Geocoordinates'],
               target="_blank",
               style={'text-decoration': 'none'}))
    # OpenStreetMaps
    # df_row.append(
    #   html.A(rows['Geocoordinates'], href='https://nominatim.openstreetmap.org/ui/search.html?q='
    #   + rows['Geocoordinates'],
    #          target="_blank",
    #          style={'text-decoration': 'none'}))
```

Fig 6.25. Codi per tractar la columna Geocoordinates de la taula d'articles de Map of Geolocated Articles. En color verd el text comentat, per poder canviar entre Google Maps i OpenStreetMap. Font: Elaboració pròpia, 2021.

Mapa

Es treballa sobre el mateix DataFrame utilitzat per la taula, però seleccionant només les columnes que es representaran al mapa, de manera que s'obtenen les columnes amb el títol de l'article, l'identificador de Wikidata, les coordenades i addicionalment s'afegeix la

columna seleccionada mitjançant el filtre *Order by feature*, que representarà la mida de cada article al mapa.

Després, s'itera sobre les llengües destí, formant els noms de les columnes que emmagatzemen el títol d'aquell article en les respectives llengües, valor que estarà buit si no existeix. Aquestes dades s'utilitzen per preparar una columna extra (*Availability*) que contindrà una llista de les llengües en les quals existeix l'article origen usant la funció ***apply()*** com es veu a la Fig 6.26, que permet aplicar aquesta funció sobre totes les columnes d'un DataFrame. Després, s'eliminen les columnes amb els noms dels articles per netejar el DF.

```
langlist = []
for i, lang in enumerate(target_langs):
    langlist.append('page_title_{}'.format(i+1))
df_map[langlist]= df[langlist]
df_map['Availability'] = df_map.apply(lambda x:get_available_langs(x[langlist], target_langs), axis=1)
df_map.drop(langlist,axis='columns',inplace=True)
```

Fig 6.26. Fragment de codi per generar la columna "Availability" i netejar el DF. Font: Elaboració pròpia, 2021.

La funció ***get_available_langs()***, donat un DF amb el valors de les columnes amb el nom dels articles de les llengües destí i els codis de llengua respectius, retorna una llista amb el codi de les llengües en què aquell article origen existeix, mitjançant una senzilla iteració que afegeix el codi si troba que el títol existeix, o continua la iteració si no, tal com es veu a la Fig 6.27.

```
def get_available_langs(row, langcodes):
    available = []
    for col,lang in zip(row,langcodes):
        if col =='':
            continue #Not available
        available.append(lang)

    return available
```

Fig 6.27. Funció que retorna una llista amb el codi de llengua de les llengües en què existeix un article. Font: Elaboració pròpia, 2021.

Abans de construir la figura (el mapa), cal tenir una col·lecció de totes les combinacions de codi de llengua, que representaran les diferents traces de la figura, al mateix temps que

formaran la llegenda interactiva que permetrà filtrar per disponibilitat. Així, per les llengües destí Català i Espanyol [‘ca’ ,’es’], existeixen quatre combinacions: [],['ca'],['es'],['ca','es'] que signifiquen no disponible, només en català, només en espanyol i disponible en ambdós respectivament. Això s’aconsegueix mitjançant la funció recursiva ***combs()*** tal com es veu a la Fig 6.28.

```
def combs(a):
    if len(a) == 0:
        return [[]]
    cs = []
    for c in combs(a[1:]):
        cs += [c, c + [a[0]]]
    return cs
```

Fig 6.28. Funció per obtenir totes les combinacions d'elements d'una llista. Font: Jonathan R a Stack Overflow (https://stackoverflow.com/a/54480126/13434796), 2019.

Per construir el mapa, s’utilitza la funció ***build_map_figure_2()***, doncs és la segona aproximació al problema, ja que la primera generava tot el mapa de cop, en comptes de construir-lo de combinació en combinació, fet que no permetia adequar les dades que es mostraven a cada bombolla a les necessitats i requisits explicats al punt 6.3.5.1.

Finalment, la figura resultant llueix tal com es veu a la Fig 6.30. A la Fig 6.29 es pot veure en detall la bombolla d’un article quan es passa el ratolí per sobre.

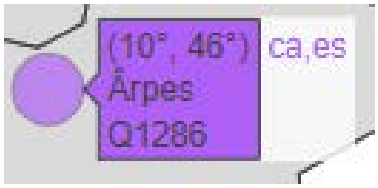


Fig 6.29. Detall de l'article sobre els Alps en la Viquipèdia Franco-provençal (frp) quan es passa el ratolí per sobre (h*over*). Es mostren les coordenades, el nom de l'article i el seu identificador de Wikidata, a més de la disponibilitat en les llengües destí. Font: Elaboració pròpia, 2021.

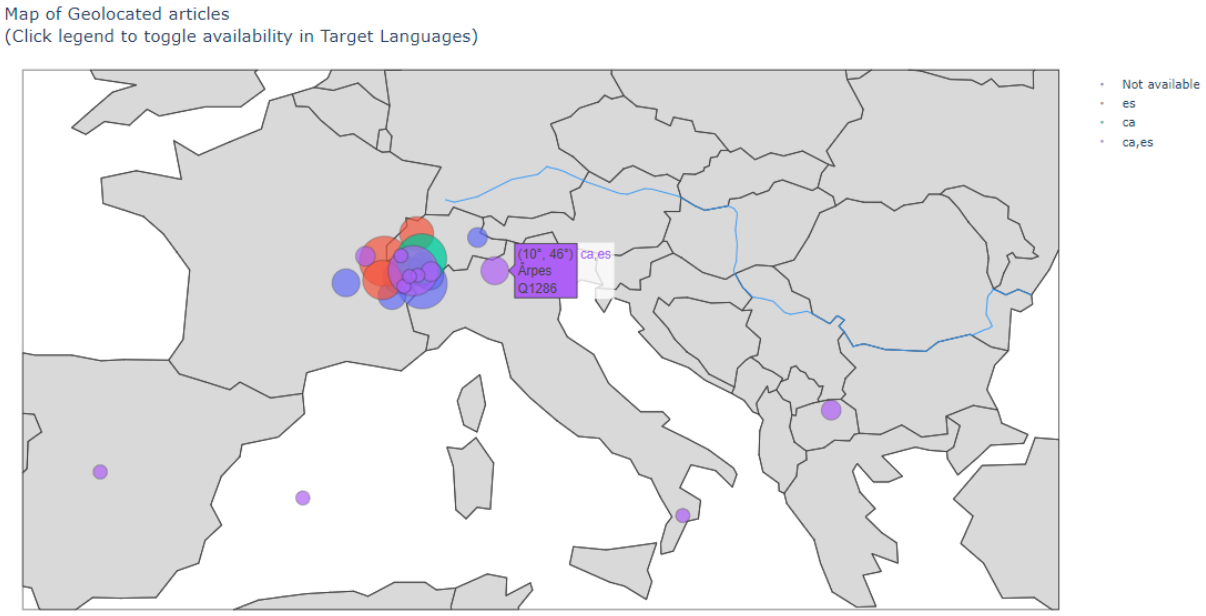


Fig 6.30. Zoom sobre el Sud d'Europa del mapa d'articles geolocalitzats sobre "Earth" en la Viquipèdia Franco-provençal (frp) i la seva disponibilitat en català (ca) i espanyol (es), ordenats per nombre de visites. Porpra per a articles disponibles en ambdues llengües, vermell per només espanyol, verd per només català i blau per no disponible ni en català ni en espanyol. Font: Elaboració pròpia, 2021.

7. Possibles ampliacions

En aquest apartat es discuteixen possibles ampliacions o aspectes de millora del projecte, sigui per part del projectista o per part de tercers, gràcies al caràcter Open Source del mateix i de l'objecte del projecte.

7.1. Generalitats

En general, cal millorar l'assignació de noms i seguir un estàndard fixe de nomenclatura pel que fa al codi i als arxius.

Pel que fa a les solucions, cal optimitzar-les de forma exhaustiva per millorar el temps de resposta, ja que per un web modern, el temps que triga en donar resposta és inacceptable. En el seu defecte, caldria donar *feedback* a l'usuari de que realment s'està realitzant la cerca amb els paràmetres desitjats amb una barra de progrés o similar.

En part aquest elevat temps de resposta, es deu a la necessitat d'aprofitar codi existent per optimitzar el procés de producció, tot i que a la llarga pugui ser contraproduent si hi ha molta propagació de negligències d'optimització o similars.

A més, es imperatiu utilitzar un mecanisme de control de versions com GIT al WDO, ja que és complicat desenvolupar sense totes les funcionalitats i facilitats que aquest aporta.

7.2. Homepage Gender Visibility

En aquesta visualització es pot utilitzar l'element de marca de temps o *timestamp* emmagatzemat a la BD per realitzar un gràfic històric rellevant quan hi hagi suficients dades per poder veure l'evolució en el temps de la diversitat, en comptes de només comptabilitzar les ocurrències.

També es pot afegir una llista de persones, amb el seu enllaç a l'element de Wikidata, que apareixen a la portada donada una versió lingüística, utilitzant l'identificador de la columna *person* de la taula *persons*, que podria estar subjecte també a la variable temporal.

A més, cal millorar la manera d'obtenir l'identificador ***page_id*** de la portada de cada Viquipèdia, doncs en algunes edicions lingüístiques aquest no és correcte i, per tant, les dades en aquests casos no són reals.

7.3. Filtres de cerca de les eines

Es pot modificar el filtre *Topic* explicat al subapartat 6.3.3.3 perquè admeti múltiples valors, fet que estalviaria crear col·leccions d'articles de cada tema per separat. Per exemple, un usuari podria seleccionar 1) Food, 2) Glam i 3) Books, i només caldria modificar la funció que aplica els paràmetres escollits a la crida SQL, afegint operadors AND per concatenar temàtiques. Així, es podria seleccionar els articles que compleixin tots els requisits, en l'exemple, llistar els articles en la llengua origen que siguin sobre menjar, GLAM i llibres.

7.4. Map of Geolocated Articles

Pel que fa al mapa, es pot millorar la llegenda, fent-la més atractiva per l'usuari i més amigable, amb filtres per poder escurçar el mapa a una regió concreta, com per exemple Europa i que només apareguin a la llista els articles que tinguin la seva localització dins de la regió. També permetria tenir més detalls relatius a la geografia de la zona com la distribució política en estats, comarques, etc. O bé utilitzar un altre tipus de mapa més interactiu per dibuixar les bombolles que representen els articles.

A més, una funcionalitat molt rellevant i còmode, seria afegir enllaços a cada bombolla, ja sigui per redirigir l'usuari a la taula amb totes les estadístiques o directament a la pàgina de Wikidata d'aquell element geolocalitzat.

7.5. Medical Articles i Monuments and Buildings Articles

En ser dues solucions pràcticament idèntiques, s'analitzaran de forma conjunta.

Es podria millorar el filtre de cerca, de manera que es pogués filtrar per tipus d'edifici o monument i per tema pertanyent a medicina per tal d'escurçar la cerca i satisfer les necessitats d'un editor molt expert en un àmbit molt concret. A més, la millora del temps d'execució de la cerca, com s'ha mencionat anteriorment, és un aspecte prioritari perquè l'usuari no marxi o es pensi que funciona malament.

8. Conclusions

Per acabar, en aquest capítol es tracta la visió general extreta de la realització del projecte, un cop acabat. Primer s'adrecen les limitacions que hi ha hagut a l'hora de desenvolupar-lo, seguit per una discussió sobre el producte i l'ús que poden donar-li els *stakeholders*. Finalment, a les conclusions finals, es parla sobre l'assoliment o no dels objectius finals i una recapitulació del treball realitzat.

8.1. Limitacions

El desenvolupament del producte s'ha vist limitat per diversos factors tant tècnics com socioemocionals. Per les possibles millores i ampliacions del producte, vegeu el capítol 7.

8.1.1. Limitacions tècniques

Primer, ha calgut adaptar-se a un codi aliè ja existent, fet que ha suposat una corba d'aprenentatge considerable, ja que és un codi poc estructurat, sense comentaris ni documentació i amb alguna mala pràctica pel que fa a l'escriptura de codi. Així i tot, ha sigut un codi aprofitable pel projecte, encara que no s'ha pogut millorar, estructurar i optimitzar per manca de temps i per sortir-se de l'abast del projecte.

Pel que fa al llenguatge de programació, al servidor hi ha la versió 3.5 de Python, mentre que actualment hi ha la 3.9 al mercat, amb més funcionalitats i correcció d'errors. Només es disposava de permisos d'escriptura al directori *dash_apps_dev* i a l'*script* per posar en marxa el web en versió *_dev*. Per tant, l'ús de llibreries que no estiguessin importades ja al *virtual environment* (*venv)* no es contemplava. Disposar d'alguna llibreria extra hagués estalviat temps en algunes fases del desenvolupament.

A més, el servidor del WDO ofereix certes limitacions pel que fa a l'emmagatzematge i la capacitat de processament, amb 8 CPU virtuals, 16GB de RAM i 160GB de disc dur, que emmagatzemant tantes BD i processos actius que les van omplint gairebé el 100% del temps, aquestes prestacions es queden molt curtes per anar ampliant funcionalitats com es veu a la Fig 8.1.

Fig 8.1. *Pie chart* de l'utilització dels recursos del WDO. Font: Pàgina del WDO a Horizon, 2021

Pel que fa a les dades, ha sigut tot el contrari, doncs la BD ***wikipedia_diversity.db*** conté moltes variables de diversitat a analitzar i es compta amb altres BD per fer interseccions i extreure més valor a aquestes.

La limitació més important ha sigut el no disposar d'un control de versions, ja que el codi que es troba al repositori de *GitHub* del WDO, és una còpia, però no s'actualitza en temps real i no permet desenvolupar mitjançant diferents branques ni tornar a versions anteriors, fet que endarrereix molt el desenvolupament pel que fa al testing. A més no tenir control de versions és un aspecte molt propens a la pèrdua d'informació.

Finalment, tot i aquestes limitacions, s'ha pogut desenvolupar el producte dins dels terminis, malgrat en certes ocasions s'hagi endarrerit, gràcies a que el codi ja existent era escalable i ampliable fàcilment encara que tingui mancances de llegibilitat i estructuració.

8.1.2. Limitacions socioemocionals

La pandèmia mundial de la COVID-19 ha afectat la producció greument, ja que ha provocat el confinament i aïllament social del projectista, que ha vist afectada la seva salut mental.

Els efectes d'aquest aïllament i incertesa, han provocat molta angoixa i sensació d'inhabilitat, que s'ha traduït en la incapacitat de, en certes èpoques al llarg del projecte, continuar treballant tant en la memòria com en el producte.

Cal remarcar que el projectista ha hagut de dur a terme dos Treballs Finals de Grau a la vegada per motius econòmics, i malgrat aquest projecte ha sortit endavant, l'altre TFG ha vist molt entorpida la seva producció, doncs les fites plantejades al document de l'Estudi de

Viabilitat, tenien en compte la producció d'ambdós treballs a la vegada. Per aquest motiu s'ha pogut acabar aquest projecte abans del temps d'entrega tot i la incapacitat de treballar en certs moments.

8.2. Discussió

El producte ha sigut l'esperat dins l'abast del projecte, però en ser una eina, no serveix de res si aquesta no s'utilitza o no s'utilitza en benefici del Moviment Wikimedia. Per tant, d'una banda cal presentar-lo als grups d'interès i explicar les funcionalitats i problemàtiques que pot ajudar a resoldre.

D'altra banda cal recalcar la importància que té desenvolupar eines amb un codi net, documentat i ben optimitzat, malgrat el producte no assoleixi aquests requisits, per tal que el Moviment Wikimedia creixi i estigui més a prop del seu objectiu, ja que si algun usuari de l'eina troba a faltar una funcionalitat, la pugui desenvolupar sense barreres tecnològiques o de llengua i es pugui treballar de manera col·laborativa, en comptes de competitiva.

Per solucionar aquests dos temes, es vol presentar el projecte al congrés internacional del Moviment Wikimedia, on es realitzen conferències per presentar estudis, investigacions, observacions i experiments relacionats amb el projecte, la cultura i la tecnologia *wiki* i el coneixement lliure: **Wikimania 2021**. D'aquesta manera, es dona a conèixer el projecte i les noves funcionalitats que té alhora que es convida a tothom a col·laborar i estendre funcionalitats o millorar i netejar l'estructura del mateix, a més d'encoratjar altres usuaris a realitzar projectes semblants, especialment altres estudiants universitaris que hagin de desenvolupar un projecte d'aquest abast.

8.3. Conclusions finals

Els objectius principals s'han assolit tant l'**OP1** com l'**OP2**, ja que s'han desenvolupat quatre solucions per omplir forats de contingut, tant utilitzant dades ja processades pel WDO com creant BD pròpies. Pel que fa al desenvolupament de la infraestructura web, en haver sigut una extensió del WDO, es pot concloure que malgrat no haver-se desenvolupat des de zero, s'ha assolit l'objectiu de visualitzar les dades processades i facilitar l'accés als usuaris.

Relatiu als objectius secundaris, s'ha pogut crear el producte tot seguint la filosofia Open Source, alineat amb els objectius del Moviment, fet que ha facilitat enormement el desenvolupament, ja que hi havia molta informació, documentació i resolució de problemes, al respecte de les tecnologies usades, disponible, malgrat l'SPARQL ha representat un repte major respecte les altres. A més, s'ha entès com desenvolupar una eina dins de Meta-Wiki i els aspectes que l'envolten, treballant amb dades en temps real utilitzant l'API de Wikimedia i en menys mesura, les rèpliques de les BD en producció de cada Viquipèdia.

Desafortunadament, no s'ha aprofundit en el disseny d'una interfície especialment amigable amb un bon estudi d'experiència d'usuari. Així i tot, es creu suficient com perquè un usuari amb bona comprensió lectora i amb un mínim d'experiència en l'ús de filtres de cerca no tingui problemes per poder utilitzar el producte de forma satisfactòria. Tampoc s'ha realitzat la difusió projectada als objectius secundaris, però com s'explica a l'apartat 8.2, es vol presentar el projecte a Wikimania 2021 per abordar aquests objectius i donar continuïtat al projecte.

Finalment, recuperant el que s'explica a la Introducció, aquest projecte no només serveix el projectista per la necessitat de realitzar un TFG, sinó que beneficia tota la població, doncs ajuda a millorar una enciclopèdia que s'utilitza cada dia arreu del món i aporta coneixement lliure, obert i gratuït, permetent que tot tipus d'usuari entenguin un article de forma més o menys profunda, segons les seves capacitats i la complexitat de l'article en qüestió.

Amb tot això, l'accés al coneixement de forma lliure i gratuïta garanteix persones més formades que puguin realitzar avenços tant culturals com científics i alhora millorar la Viquipèdia, creant una bola de neu on quant més coneixement hi ha, més se n'afegeix, així millorant la vida a la Terra en cada iteració.

Per aquesta raó, malgrat els inconvenients provocats per la COVID-19, ha valgut la pena dur a terme aquest projecte en tots els seus aspectes i s'espera que aquest document encoratgi altres persones a executar projectes similars.

9. Bibliografia

[1] «Alexa - Top sites». https://www.alexa.com/topsites (consulta gen. 05, 2021).

[2] C. Okoli, «A Brief Review of Studies of Wikipedia in Peer-Reviewed Journals», en *2009 Third International Conference on Digital Society*, feb. 2009, p. 155-160. doi: 10.1109/ICDS.2009.28.

[3] «Research:Content gaps on Wikipedia». https://meta.wikimedia.org/wiki/Research:Content_gaps_on_Wikipedia (consulta març 02, 2021).

[4] M. R. Laurent i T. J. Vickers, «Seeking Health Information Online: Does Wikipedia Matter?», *Journal of the American Medical Informatics Association*, vol. 16, núm. 4, p. 471-479, jul. 2009, doi: 10.1197/jamia.M3059.

[5] «Wikipedia Founder Jimmy Wales Responds - Slashdot». https://slashdot.org/story/04/07/28/1351230/wikipedia-founder-jimmy-wales-responds (consulta des. 21, 2020).

[6] «Portal:Toolforge/About Toolforge - Wikitech». https://wikitech.wikimedia.org/wiki/Portal:Toolforge/About_Toolforge (consulta gen. 20, 2021).

[7] A. Halfaker i J. Riedl, «Bots and Cyborgs: Wikipedia's Immune System», *Computer*, vol. 45, núm. 03, p. 79-82, març 2012, doi: 10.1109/MC.2012.82.

[8] L. (Nico) Zheng, C. M. Albano, N. M. Vora, F. Mai, i J. V. Nickerson, «The Roles Bots Play in Wikipedia», *Proc. ACM Hum.-Comput. Interact.*, vol. 3, núm. CSCW, p. 215:1-215:20, nov. 2019, doi: 10.1145/3359317.

[9] «Dashboard (business)», *Wikipedia*. des. 26, 2020. Consulta: gen. 26, 2021. [En línia]. Disponible a: https://en.wikipedia.org/w/index.php?title=Dashboard_(business)&oldid=996334463

[10] S. Few, *Information dashboard design: the effective visual communication of data*, 1st ed. Beijing ; Cambride [MA]: O'Reilly, 2006.

[11] M. Miquel-Ribé i D. Laniado, «The Wikipedia Diversity Observatory: A Project to Identify and Bridge Content Gaps in Wikipedia», en *Proceedings of the 16th International Symposium on Open Collaboration*, Virtual conference Spain, ago. 2020, p. 1-4. doi: 10.1145/3412569.3412866.

[12] M. Miquel-Ribé i D. Laniado, «Wikipedia Cultural Diversity Dataset: A Complete Cartography for 300 Language Editions», *ICWSM*, vol. 13, p. 620-629, jul. 2019.

[13] «Wikipedia Diversity Observatory - Meta». https://meta.wikimedia.org/wiki/Wikipedia_Diversity_Observatory (consulta juny 05, 2021).

[14] «Wikipedia:What Wikipedia is not», *Wikipedia*. feb. 03, 2021. Consulta: feb. 04, 2021. [En línia]. Disponible a: https://en.wikipedia.org/w/index.php?title=Wikipedia:What_Wikipedia_is_not&oldid=1004661803

[15] «Wikipedia:Five pillars», *Wikipedia*. gen. 23, 2021. Consulta: feb. 04, 2021. [En línia]. Disponible a: https://en.wikipedia.org/w/index.php?title=Wikipedia:Five_pillars&oldid=1002269078

[16] «Wikidata:Introduction - Wikidata». https://www.wikidata.org/wiki/Wikidata:Introduction (consulta març 05, 2021).

[17] P. Massa i F. Scrinzi, «Manypedia: Comparing language points of view of Wikipedia communities», *First Monday*, vol. 18, gen. 2013, doi: 10.5210/fm.v18i1.3939.

[18] E. Borra *et al.*, «Societal Controversies in Wikipedia Articles», en *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, Seoul, Republic of Korea, abr. 2015, p. 193-196. doi: 10.1145/2702123.2702436.

[19] «PEP 333 -- Python Web Server Gateway Interface v1.0», *Python.org*. https://www.python.org/dev/peps/pep-0333/ (consulta maig 04, 2021).

[20] «Python Developers Survey 2018 Results», *JetBrains*. https://www.jetbrains.com/research/python-developers-survey-2018/ (consulta gen. 25, 2021).

[21] «PYPL PopularitY of Programming Language index». https://pypl.*Github*.io/PYPL.html (consulta gen. 25, 2021).

[22] Kostis C. KOUTSOPOULOS, «CHANGING PARADIGMS OF GEOGRAPHY», *European Journal of Geography*, núm. 1, p. 54-75, 2011.

[23] R. M. Downs, «The Need for Research in Geography Education: It Would be Nice to Have Some Data», *Journal of Geography*, vol. 93, núm. 1, p. 57-60, gen. 1994, doi: 10.1080/00221349408979690.

[24] J. M. Heilman *et al.*, «Wikipedia: A Key Tool for Global Public Health Promotion», *Journal of Medical Internet Research*, vol. 13, núm. 1, p. e14, 2011, doi: 10.2196/jmir.1589.

[25] Y. Meta contributors, «Wiki World Heritage User Group - Meta», *Viquipèdia, l'enciclopèdia lliure*. juny 02, 2021. Consulta: juny 02, 2021. [En línia]. Disponible a: https://meta.wikimedia.org/w/index.php?title=Wiki_World_Heritage_User_Group&oldid=21511143

[26] «Wikipedia:Multiyear ranking of most viewed pages», *Wikipedia*. abr. 13, 2021. Consulta: abr. 20, 2021. [En línia]. Disponible a: https://en.wikipedia.org/w/index.php?title=Wikipedia:Multiyear_ranking_of_most_viewed_pages&oldid=1017580747