



Treball de Fi de Grau

GRAU D'ENGINYERIA INFORMÀTICA

**Facultat de Matemàtiques
Universitat de Barcelona**

**DISSENY D'UNA XARXA SOCIAL WEB PER A
PODER COMPARTIR NOTÍCIES**

David Ortega Lázaro

Directora:

Dra. Maria Salamó Llorente

Realitzat a:

Departament de Matemàtica Aplicada i Anàlisi
Universitat de Barcelona

Barcelona, 19 de juny de 2014

Resum

El projecte que es desenvoluparà en aquest Treball de Fi de Grau és una aplicació web que té característiques tant de xarxa social com de mitjà de comunicació convencional. El projecte defineix, dissenya i implementa una xarxa social web per a poder compartir notícies de forma fàcil, amigable i amb una alta usabilitat, on els usuaris puguin autogestionar-se la informació a compartir i rebre. Visualment els usuaris podran interactuar amb les notícies, les podran veure, podran redactar noves, podran comentar-les i a més les podran visualitzar en un gran mapa veient els punts més actius del planeta.

Estem acostumats a veure com a diari se'ns informa d'una manera molt particular de tots els fets que ocorren al món, donant una visió normalment esbiaixada. Molts diaris viuen de subvencions polítiques, sobreviuen gràcies als que estan al poder, i tenen la necessitat de donar el seu suport sempre, fins i tot quan van en contra dels interessos del poble en general.

Per aquest motiu neix aquesta aplicació, per poder explicar de primera mà el que ens envolta, sense intervencions polítiques ni ideològiques, des del punt de vista de qui ho viu, possiblement una de les majors fonts de veracitat.

Una aplicació on el poble informa del que passa al poble.

Aquesta aplicació web també disposa d'una aplicació de suport per a telèfons intel·ligents amb el sistema operatiu Android que es realitza en un altre Treball de Fi de Grau.

Resumen

El proyecto que se desarrollará en este Trabajo de Fin de Grado es una aplicación web que tiene características tanto de red social como medio de comunicación convencional. El proyecto define, diseña e implementa una red social web para poder compartir noticias de forma fácil, amigable y con una alta usabilidad, donde los usuarios puedan autogestionarse la información a compartir y recibir. Visualmente los usuarios podrán interactuar con las noticias, las podrán ver, podrán redactar nuevas, podrán comentarlas y además las podrán visualizar en un gran mapa viendo los puntos más activos del planeta.

Estamos acostumbrados a ver cómo a diario se nos informa de una manera muy particular de todos los hechos que ocurren en el mundo, dando una visión normalmente sesgada. Muchos diarios viven de subvenciones políticas, sobreviven gracias a aquellos que están en el poder, y tienen la necesidad de dar su apoyo siempre, incluso cuando van en contra de los intereses del pueblo en general.

Por este motivo nace esta aplicación, para poder explicar de primera mano lo que nos rodea, sin intervenciones políticas ni ideológicas, desde el punto de vista de quien lo vive, posiblemente una de las mayores fuentes de veracidad.

Una aplicación donde el pueblo informa de lo que ocurre en el pueblo.

Esta aplicación web cuenta además con una aplicación de soporte para teléfonos inteligentes con el sistema operativo Android que se realiza en otro Trabajo de Fin de Grado.

Abstract

The project that will be developed in this End-of-Degree Project is a web application that counts with features on both social networks and conventional journals. The project defines, designs and implements a social networking site for sharing news in an easy and friendly way and with high usability, where users can self-managing information sharing and receiving. Visually users can interact with the news, they can see them, they can write new comments and they will be able to view the news on a big map to see the most active places in the world.

Nowadays, we are used to see how every newspaper provides a very subjective point of view of every single piece of news that is published. Therefore, this gives us a particular perception of the original fact, which is usually manipulated. Countless newspapers survive thanks to political subsidies, as they are committed to deliver its support to political leaders, even if they go against ordinary citizens' interests.

For that reason, this application is designed to give first-hand information directly from the original source, to put it simple, from those currently living the facts, as they are possibly the major base of accuracy and reliability.

An application where citizens inform of what happens in the city.

This web application has also an Android mobile application for smartphones, which is designed in other End-of-Degree Project.

INDEX

Resum	2
Resumen	3
Abstract.....	4
1 - Introducció	7
1.1 - Motivació.....	7
1.2 - Objectius.....	7
2 - Estudi de mercat.....	9
2.1 - Aplicacions similars	9
2.1.1 - Menéame.....	9
2.1.2 - La Tafari	10
2.1.3 - Synkratos	10
2.2 - Missió i visió	11
2.3 - Anàlisis dels navegadors.....	11
2.4 - Anàlisis DAFO	12
2.4.1 - Debilitats	12
2.4.2 - Amenaces.....	12
2.4.3 - Fortaleses.....	12
2.4.4 - Oportunitats	12
3 - Arquitectura.....	13
3.1 - Llenguatges	13
3.1.1 - HTML.....	13
3.1.2 - CSS	14
3.1.3 - JavaScript	14
3.1.4 - Python	15
3.1.5 - SQL	15
3.2 - Django	16
3.2.1 - URL dispatcher	16
3.2.2 - Views	17
3.2.3 - Templates.....	17
3.2.4 - Models.....	17
4 - Anàlisi i disseny.....	19
4.1 - Requeriments	19
4.2 - Casos d'ús	19
4.2.1 - Diagrama de casos d'ús	19
4.2.2 - Descripció de casos d'ús textuals	20
4.3 - Disseny de la base de dades.....	24
4.3.1 - Model ER.....	24
4.3.2 - Descripció de taules.....	25
5 - Desenvolupament	28
5.1 - Entorns de desenvolupament utilitzats.....	28
5.1.1 - OpenShift	28
5.1.2 - GitHub	28
5.2 - Programació	29

5.2.1 - Arxius Python (Controlador i Model)	29
5.2.2 - Plantilles HTML (Vista).....	33
6.2.3 - Arxius estàtics (Vista)	37
5.3 - API de Google Maps	40
6 - Resultats i test	41
6.1 - Metodologia.....	41
6.2 - Resultats	42
7 - Estudi econòmic	46
8 - Conclusions i treball futur.....	48
9 - Referències bibliogràfiques	50
10 - Annex	53
10.1 - Manual d'instal·lació	53
10.2 - Manual d'usuari	54

1 - Introducció

En aquest projecte es busca definir, dissenyar i implementar una xarxa social de notícies on gent de tot arreu pugui publicar de manera lliure i personal.

L'aplicació web que es desenvolupa en aquest treball té moltes similituds amb les xarxes socials que estan tant de moda a l'actualitat i aprofita un dels seus principals usos, el poder compartir les notícies d'arreu del món, per tal de fer una aplicació nova que pugui ser utilitzada de manera real per persones de tot arreu sense importar els seus coneixements tècnics.

L'aplicació està enfocada a permetre una visió més personal i propera a la notícia que la que li donen els grans mitjans de comunicació. Permet, per tant, que aquestes notícies estiguin explicades a través de les paraules dels seus protagonistes, sense intermediaris que alterin la informació.

Al dia a dia estem acostumats a veure com diferents diaris, diferents canals de televisió, diferents emissores de ràdio, etc, donen sempre la mateixa notícia. Però hi ha una cosa que sempre tenen aquests mitjans de comunicació, i és que aquesta notícia comuna sempre estarà explicada d'una manera totalment diferent, podent trobar diaris amb la mateixa notícia explicant coses contràries sense que el lector pugui saber què passa realment.

Fer un nou mitjà de comunicació que sigui lliure, independent de línies editorials, independent de qualsevol ideologia és una tasca molt difícil, però en aquest treball es busca aconseguir aquest objectiu.

1.1 - Motivació

La motivació és fer un projecte des de 0, a la vegada que s'utilitzaran els coneixements adquirits a la carrera.

Poder fer una web de manera íntegra ha estat sempre un dels meus objectius personals, però que fins ara mai havia pogut dur a terme. En aquest projecte veig com uns dels meus objectius es compleix, motiu amb molt de pes a la hora d'haver escollit la temàtica del Treball de Fi de Grau.

Podem resumir doncs la motivació en fer alguna cosa que sigui novedosa i innovadora i que a més aporti caire social.

1.2 - Objectius

L'aplicació té com a objectiu ser una xarxa social de referència en el seu àmbit, ja que es tracta d'una xarxa social dedicada íntegrament i de manera gaire bé exclusiva a les notícies, diferenciant-se de les xarxes socials principals que tenen com a objectiu connectar a les persones d'una manera més propera i sense fer tant d'èmfasi a les notícies.

L'objectiu principal del projecte és el de fer una xarxa social. Aquest objectiu es complementa, a més, amb una sèrie de subobjectius tals com poder permetre l'accés controlat a la xarxa social, poder fer que els usuaris publiquin notícies, puguin cercar-les, comentar-les, etc. A més, podran veure en un mapa a pantalla completa quins són els punts del planeta més actius, on hi ha més notícies en un moment determinat.

Fer que un usuari sense tenir coneixements de caràcter tècnic ni avançats pugui fer servir aquesta xarxa social sense cap complicació a través d'una interfície intuïtiva i molt visual és també un altre objectiu a tenir en compte, ja que volem normalitzar i familiaritzar a l'usuari a l'ús habitual de l'aplicació web.

A més a més, aquest projecte va de la mà d'una altre Treball de Fi de Grau, que consistirà en fer una aplicació per a telèfons intel·ligents amb el Sistema Operatiu *Android* que es comuniqui amb l'aplicació web desenvolupada a partir d'aquest projecte. En aquest sentit, els objectius mencionats que fan referència a la consecució de poder crear un nou mitjà de comunicació són compartits amb aquest altre treball paral·lel.

2 - Estudi de mercat

En aquest capítol es fa un estudi de mercat de les diferents xarxes socials que existeixen avui en dia.

2.1 - Aplicacions similars

Tot i que el resultat final del desenvolupament d'aquesta aplicació web sigui una xarxa social més, aquesta no és com les altres. La diferència principal és que les xarxes socials convencionals estan enfocades a un caràcter estrictament social i de lleure, essent xarxes on els seus usuaris expliquen el seu dia a dia a través de missatges curts o de fotografies i vídeos sense cap motivació més enllà de la de poder compartir la seva vida i les seves aficions pel simple plaer de comunicar-se amb els seus amics, companys de feina, familiars o persones amb les que comparteixen gustos.

A més a més, l'aplicació també té similituds amb els fòrums d'opinions que parlen de temes del dia a dia o simplement de temes concrets dedicats a un àmbit específic. Per altra banda, la xarxa està plena de blocs on els seus actors poden donar la seva opinió de l'actualitat sense que ningú no intervingui.

En qualsevol cas, els blocs estan molt limitats, ja que has d'anar un per un sense tenir la possibilitat de tenir un espai comú on poder interactuar amb tots a la vegada de manera ràpida, fàcil i eficient.

Aquesta aplicació, però, té una gran diferència a les pàgines webs i aplicacions abans mencionades, està dedicada de manera exclusiva a compartir notícies, i més específicament, a notícies locals, per tal de donar-les a conèixer en el món sencer sense intermediaris ni grups mediàtics pressionant.

Tot i així, després d'un treball d'investigació dut a terme durant els mesos anteriors, i també durant el transcurs del desenvolupament del projecte, hem trobat projectes similars que coincideixen amb l'aplicació que es vol desenvolupar.

2.1.1 - Menéame

Menéame^[1] és una pàgina web que també fa la funció de xarxa social per compartir notícies. Però té una gran diferència amb l'aplicació d'aquest projecte, i és que les notícies provenen d'altres mitjans de comunicació, per tant es perd aquesta llibertat informativa.

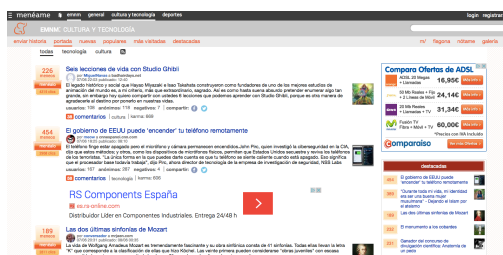


Fig. 1: Pàgina inicial de Menéame

A més no permet veure en cap moment en un mapa les notícies d'arreu del món. Es tracta, per tant, d'un afegidor de notícies, i no d'una font de notícies noves.

2.1.2 - La Tafari

De la mateixa forma que *Menéame* és un afegidor de notícies, **La Tafari**^[2] és exactament el mateix tipus d'aplicació amb la diferència que està feta a Catalunya i les seves notícies provenen de fonts d'informació en català.

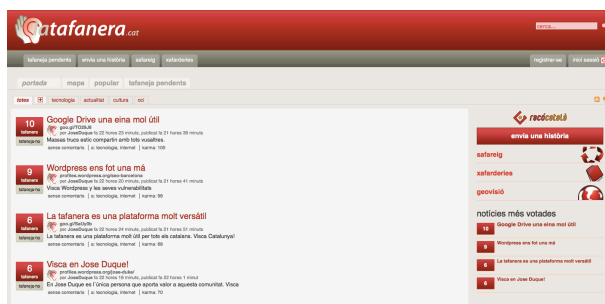


Fig. 2: Pàgina inicial de La Tafari

A part d'aquests dos llocs webs esmentats, existeixen multitud d'aplicacions que fan les mateixes funcions d'una manera similar, i tots aquests projectes s'allunyen molt del que volem aconseguir en aquest Treball de Fi de Grau: poder redactar notícies locals des del punt de vista del protagonista i donar difusió global a les mateixes.

2.1.3 - Synkratos

De totes les aplicacions web similars que he trobat a la xarxa, **Synkratos**^[3] és la que funciona de la manera més similar a l'aplicació que vull desenvolupar. Aquesta aplicació compleix els objectius que volia complir amb el projecte d'aquest Treball de Fi de Grau, i compta a més amb una interfície gràfica molt similar a la desenvolupada en aquest projecte, però té una gran diferència que l'allunya de l'aplicació que es desenvolupa en aquest projecte.

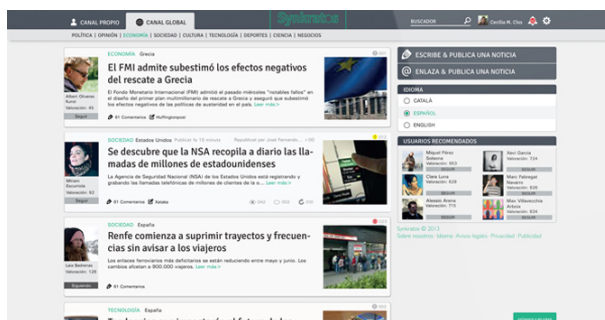


Fig. 3: Pàgina inicial de Synkratos

Synkratos no ofereix cap possibilitat de visualitzar en un mapa del món totes les notícies, permetent tenir una visió ràpida de quins són els llocs més actius informativament del planeta, mentre que la nostra aplicació sí que ho permet.

Aquest web va ser trobat quan aquest Treball de Fi de Grau estava en una fase molt avançada del desenvolupament, pràcticament al final del cicle del treball, i va resultar tenir un disseny molt similar a l'aplicació que es desenvolupa, tot i que mai abans l'havíem vist i les similituds amb el disseny han estat casuals.

2.2 - Missió i visió

En el nostre cas som un projecte que ofereix a la població un portal on pot explicar els esdeveniments des del seu punt de vista. Els hi oferim un lloc on poden saber que passa realment al món.

Volem convertir-nos en un portal que ofereixi la oportunitat a la població de poder saber la realitat que està passant al món i que relacionin el nostre producte amb un portal de coneixement on s'intercanvia informació sense ser aquesta manipulada o falsa.

2.3 - Anàlisi dels navegadors

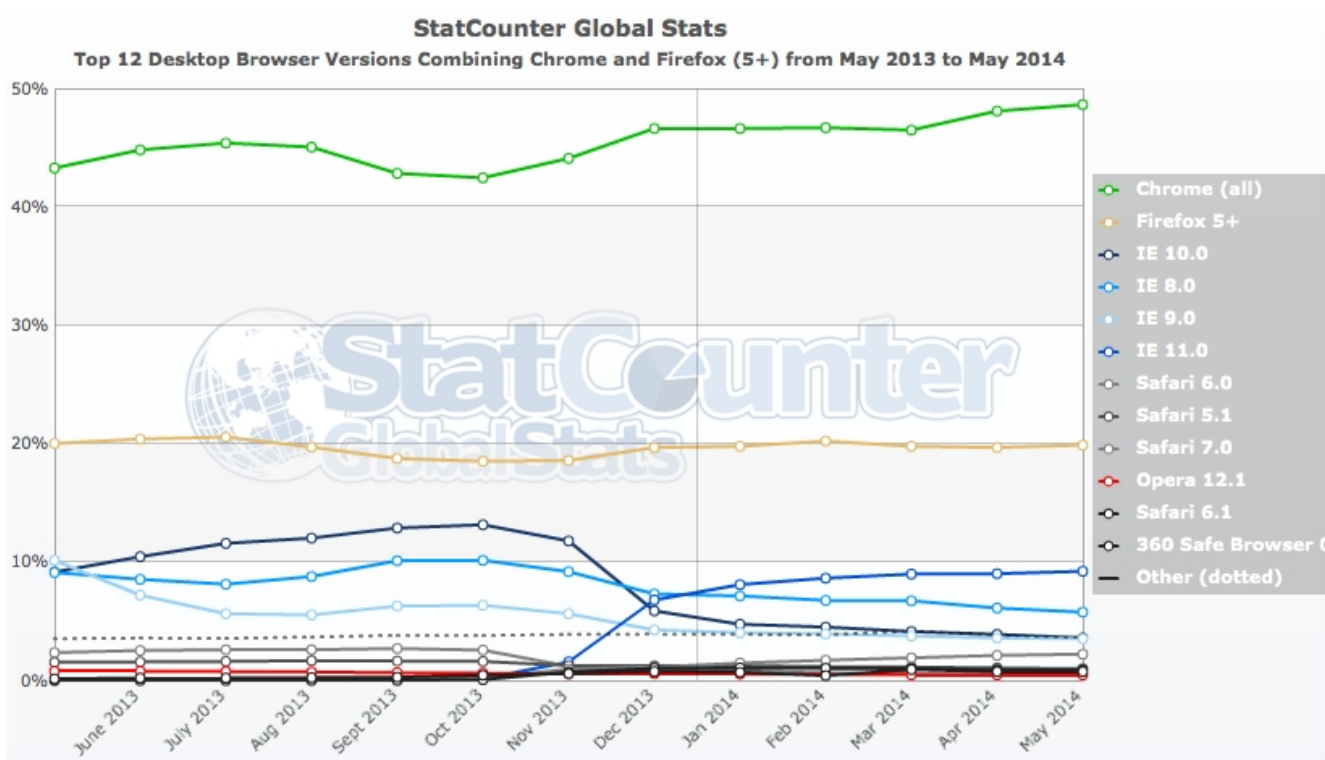


Fig. 4: Gràfic d'ús dels navegadors des de Juny de 2013 fins a Maig de 2014

Durant l'últim any, les estadístiques d'ús dels navegadors^[4] d'internet no han canviat gaire. L'únic canvi significant ha estat que Internet Explorer 10 ha disminuït la seva quota del mercat per ser substituït per la versió 11 del conegut navegador de Microsoft.

En aquest sentit, es tracta d'un fet que no ens afecta gaire, ja que com veurem més endavant, al **capítol 3**, l'ús de la versió 10 o 11 és indiferent pel funcionament correcte de la web.

La dada important és que més d'un 50% dels visitants potencials de l'aplicació web podran fer servir totes les funcions proposades a la web gràcies a fer servir navegadors compatibles amb elles: Google Chrome (**50%**) + Mozilla Firefox (**20%**) + Internet Explorer (**10%**) entre d'altres.

2.4 - Anàlisis DAFO

Un anàlisis DAFO (Debilitats-Amenaces-Fortaleses-Debilitats) és un anàlisis que fan els departaments de màrqueting de totes les empreses per poder conèixer quins són els punts forts i febles del seu producte. En aquest projecte també he volgut fer un anàlisis DAFO per poder saber quins són els punts clau del projecte per tal d'explotar-los al màxim.

2.4.1 - Debilitats

- El projecte té pocs recursos econòmics.
- Pocs treballadors.
- Ens incorporem a un mercat nou per nosaltres (el de les notícies).
- Falta d'experiència en el mercat.
- El servidor que utilitzem és poc potent i no aguanta molt de tràfic.

2.4.2 - Amenaces

- Els nostres competidors son més forts i tenen més recursos.
- Els competidors estan ben posicionats al mercat.
- Existència d'altres xarxes socials convencionals que podrien suplir la funció del projecte.
- En l'actualitat existeixen molts mitjans de comunicació.
- La competència ja té aplicacions mòbils de les seves pàgines.

2.4.3 - Fortaleses

- Cobrim una necessitat del mercat existent.
- Els nostres usuaris poden veure en un mapa quins són els punts del planeta més actius.
- Deixem que l'usuari sigui el propi autor de la notícia.
- Disposem d'una interfície clara, senzilla i simple. Molt útil per usuaris sense experiència.
- L'aplicació està dirigida a tot el segment de la població del mercat.
- Tenim una aplicació multidispositiu (web i smartphone).

2.4.4 - Oportunitats

- Hi ha una gran part de la població que reclama un lloc on poder donar la seva visió dels fets.
- Hi ha una gran part de la població que reclama un lloc on poder llegir la visió dels protagonistes de les notícies.
- Solventem una carència del mercat.

3 - Arquitectura

Durant la fase de disseny de l'aplicació web es van estudiar diversos llenguatges per poder programar la pàgina web i es van analitzar els seus inconvenients i avantatges. En un primer moment es va optar per fer servir PHP com a llenguatge de programació junt als llenguatges estàndard de HTML i CSS.

Es va escollir PHP^[5] degut a que era un llenguatge que coneixia i amb el qual ja havia treballat prèviament. Per tant suposaria fer un projecte amb un llenguatge que ja era familiar i poder explotar-lo al màxim a la vegada que aprenia coses noves sobre el mateix. PHP permet connectar-nos a la base de dades de manera senzilla i per tant no suposava cap problema a la hora de desenvolupar les funcions bàsiques de l'aplicació.

Finalment, però, PHP es va descartar, ja que com que aquest Treball de Fi de Grau va lligat a un projecte que està programat sobre Android, era necessari que la connexió entre ambdós fos simple i ràpida. Es va arribar a la conclusió de que la millor manera de realitzar aquesta connexió era a través d'un framework que les connectés, i després de mirar les diferents opcions es va optar per realitzar la web amb l'ajuda del framework Django.

Això suposava, a part de poder connectar les dues aplicacions fàcilment, aprendre una nova manera de programar per web, ja que el framework descartava fer servir PHP i significava utilitzar Python com a llenguatge de programació. Així doncs, els llenguatges utilitzats en aquest projecte han estat finalment HTML^[6], CSS^[8], JavaScript^[9], Python^[11] i SQL^[12]. El motiu de la elecció de cada llenguatge s'explica a continuació.

3.1 - Llenguatges

3.1.1 - HTML

El llenguatge més bàsic utilitzat a tota pàgina web és l'**HyperText Markup Language**, més conegut per les seves sigles **HTML**^[6]. La primera cosa que cal saber sobre l'HTML és que no és un llenguatge de programació, sinó que realment és un llenguatge per publicar, el que es coneix com a *markup language*.

Les característiques bàsiques que té l'HTML és que es tracta d'un llenguatge que serveix per organitzar el contingut de les pàgines web, on va cada cosa, que apareix a cada pàgina, etc. És un llenguatge bàsic que pràcticament l'únic que permet es fer pàgines web estàtiques, és a dir, sense programació i que sempre tinguin el mateix contingut. Per aquets motiu cal fer servir altres llenguatges auxiliars per a poder dinamitzar les pàgines^[7].

En el cas concret de l'aplicació que es desenvolupa en aquest projecte es fa servir la cinquena versió d'aquest llenguatge, coneguda com **HTML5**. El motiu concret de fer servir aquesta versió és que es tracta de la última i la gran majoria de navegadors moderns la fan servir també. A més, permet la inserció de vídeos a les pàgines sense gaire problemes i sense haver de recórrer a altres llenguatges com ara el *Flash*.

```
<!DOCTYPE HTML>
<html>
  <head>
    <title> Títol de la pàgina </title>
  </head>
  <body>
    <span class="ex"> Hello World! </span>
  </body>
</html>
```

Fig. 5A: Exemple de codi HTML

Hello World!

Fig. 5B: Sortida produïda pel codi anterior

3.1.2 - CSS

De la mateixa manera que HTML^[6] ens serveix per a poder col·locar elements a la pàgina i poder decidir el seu contingut, el llenguatge **Cascading Style Sheet**, més conegut com **CSS**^[8], ens serveix per a donar-li forma, color i altres atributs al contingut de la pàgina.

Si bé això també es pot fer amb HTML pur, el CSS ens permet poder definir *classes* que ens estalviaran haver d'escriure per cada element que sigui igual que un altre el seu estil. Diem doncs que el CSS es un llenguatge per dissenyar, el que es coneix com *style sheet language*.

En el cas concret de l'aplicació dissenyada, farem servir la tercera versió d'aquest llenguatge, el **CSS3**, degut a que és el complement natural de l'HTML5 i els navegadors moderns és el que fan servir.

```
.ex {
  font-family: Arial, sans-serif;
  font-weight: bold;
  color: red;
}
```

Fig. 6A: Exemple de classe CSS

Hello World!

Fig. 6B: Disseny aplicat per la classe "ex"

3.1.3 - JavaScript

Fins ara els dos llenguatges explicats ens permeten generar pàgines web estàtiques, sense cap tipus de programació que permetin poder interactuar. Per tant hem de fer servir un llenguatge de programació adaptat a la web. Aquí és on entra en joc el llenguatge de programació **JavaScript**^[9].

JavaScript, que no s'ha de confondre amb el llenguatge Java tot i tenir sintaxi i noms molt similars, permet que l'usuari pugui interactuar amb les pàgines web i permet modificar parts del lloc web segons les accions de l'usuari. És important destacar que no es tracta d'un llenguatge orientat a objectes, sinó que és un llenguatge de programació per events.

També és important senyalar que el codi s'executa al propi client, és a dir al navegador, i que per tant mai envia dades, si no s'especifica, al servidor. Això també permet que el servidor no es sature quan hi han moltes peticions dels clients, i permet programar d'una manera lleugera i ràpida.

```
function f(n) {  
  if (n == 1) {  
    return 1;  
  } else {  
    return 0;  
  }  
}
```

Fig. 7: Exemple de codi JavaScript

A més a més, també s'ha optat per utilitzar una llibreria molt coneguda sobre JavaScript, **jQuery**^[10], que simplifica moltes de les funcions que es programen en JavaScript^[9]. Entre les característiques principals de jQuery està que permet fer funcions escrivint menys línies de codi o amb paraules més curtes. A més, conté un conjunt de funcions que donen “mobilitat” a la web, és a dir, permeten fer aparèixer o desaparèixer certs elements de manera ràpida i fàcil.

```
document.getElementById("ex");  
$("#ex");
```

Fig. 8: A dalt una línia en JavaScript, a sota simplificada amb jQuery

3.1.4 - Python

Fins ara ja tenim una pàgina que combinant tres llenguatges de característiques diferents es complementen per tenir un lloc web amb contingut, disseny i interactiu. Però aquesta aplicació ara per ara no està connectada a cap base de dades, i les característiques de l'aplicació a desenvolupar en aquest projecte fan necessari una base de dades per encriptar les dades que els usuaris enviïn i poder permetre que cada usuari tingui un espai propi a la web.

És aquí on entra en joc el framework Django, que com ja em dit abans, substitueix la idea inicial de fer servir PHP^[5] per facilitar la connexió amb Android. Però Django en si no és cap llenguatge, sinó que és un framework que funciona sota **Python**^[11], llenguatge estudiat des del primer dia a la Facultat, i que per tant no hauria de donar gaires problemes a la hora de programar.

Un cop dit això, queda bastant clar que la elecció de Python per a programar ha estat més aviat obligatòria que no pas voluntària, ja que el simple fet de fer servir aquest framework obligava a programar en Python.

3.1.5 - SQL

Una vegada ja tenim el lloc web i el llenguatge que ens ajudarà a connectar-nos a la base de dades, i de pas també amb l'aplicació del projecte paral·lel per a Android, queda definir la base de dades que farem servir en el desenvolupament de la nostra aplicació.

Durant les primeres setmanes, a la fase de disseny i anàlisi, van ser estudiades diferents bases de dades de tipus diferents. Finalment es va optar per fer servir **MySQL**^[12], que com el seu nom indica està basada en el llenguatge **Structured Query Language**, més conegut com **SQL**.

El principal motiu d'escollir aquesta plataforma ha estat bàsicament que és la més utilitzada a tot la xarxa i a més fa servir un llenguatge que ja coneixíem d'abans de l'assignatura de *Bases de Dades*. Així doncs, aquesta elecció es va prendre per consens entre els dos projectes, ja que la part de la Bases de Dades es compartida pels dos treballs finals de grau.

3.2 - Django

Fins aquí ja tenim la web funcionant de manera dinàmica i tenim feta la connexió amb la base de dades per a poder guardar les notícies, les dades dels usuaris, les fotografies i vídeos que vulguin penjar i tota mena de dades importants pel funcionament correcte de l'aplicació web. Però ara necessitem poder fer la connexió amb Android per poder comunicar ambdós projectes i a més necessitem un framework per a poder utilitzar Python^[11] a la xarxa.

Després de valorar diverses possibilitats, vam decidir escollir **Django**^[13], que tenia totes les característiques que necessitàvem i a més tenim referents d'altres persones que ja havien treballat amb ell que ens ho van recomanar pel seu fàcil funcionament i totes les facilitats que donava a la hora de desenvolupar un projecte web.

Durant les primeres setmanes de la fase de disseny, moltes hores van estar dedicades a llegir cada pàgina del manual de Django^[14], llegint les seves funcionalitats, les seves apis, les seves característiques, tot.

Ara anem, però, a analitzar les característiques més importants de Django i per què hem utilitzat cadascuna en el projecte. Tres són els passos bàsics que dona Django per mostrar cada pàgina de la web. Més endavant, al **capítol 5**, explicaré detalladament aquests passos i els documents que els componen. Ara explicaré de manera breu què són aquests tres passos i que conformen el pilar de Django.

Un detall a destacar és que els dos projectes funcionen sota **Django 1.6**, ja que aquesta versió era la última disponible quan es va començar a desenvolupar l'aplicació web i l'aplicació Android.

En aquest esquema podem veure com es comuniquen aquests tres pilars bàsics:



Fig. 9: Comunicació entre els diferents mòduls de Django

3.2.1 - URL dispatcher

Com tota pàgina web, l'accés a les diferents funcions es fa mitjançant diferents URLs que hem d'introduir al navegador per poder entrar. Django^[13] té definit en un arxiu especial, o més, aquestes URLs i les enllaça al segon pilar funcional de Django, les vistes.


```
urlpatterns = patterns('',
    url(r'^$', home),
    url(r'^admin/',
    url(r'^compose/$', compose),
    url(r'^error404/$', error404),
)
```

Fig. 10: Exemple de URL dispatcher

3.2.2 - Views

Les URLs redireccionen als visitants a les vistes, o views. Una view és bàsicament una funció que crea un contingut dinàmic que després serà mostrat finalment a l'usuari a la pàgina a la que accedeixi.

Aquestes funcions poden tenir operacions com per exemple agafar contingut de la base de dades, fer càlculs, inserta continguts a la base de dades, eliminar contingut, etc. O poden simplement mostrar una pàgina estàtica definida prèviament, o no, a l'usuari. Més endavant explicaré detalladament les views del projecte d'aquest treball.

```
def mapa(request):
    return render(request, 'map.html')
```

Fig. 11: Exemple de vista que mostra una pàgina estàtica

Tot i així, també poden haver vistes que funcionen com a funcions pròpiament dites i no mostren cap pàgina a l'usuari. Es poden utilitzar doncs per fer funcions de suport a altres vistes.

3.2.3 - Templates

Finalment, el tercer pas que efectua Django abans de mostrar una pàgina a l'usuari són les plantilles, o templates. Els templates són uns arxius que, a diferència dels altres dos tipus de funcions esmentades, estan escrits en HTML^[6].

Però tenen una característica que les diferencia de les pàgines HTML normals, i es que poden tenir etiquetes especials creades per Django^[13] per tal de dinamitzar el contingut que apareix a les pàgines.

```
[...]
<div id="leftbar">
  <div id="news">
    {% for noticia in noticies %}
[...]
```

Fig. 12: Part d'una plantilla, on podem veure una etiqueta de Django (en vermell) i etiquetes HTML

3.2.4 - Models

A més dels tres pilars explicats anteriorment, Django^[13] també té altres tipus de funcions, però les més destacables són les que permeten simplificar de manera molt significativa la definició de taules i la comunicació amb la base de dades, els models. Més endavant, al **capítol 5**, explicaré amb més detall els models que hem fet servir.

```
class Comentari(models.Model):
    noticia = models.ForeignKey(Noticia)
    contingut = models.TextField()
    autor = models.ForeignKey(User)
    data = models.DateTimeField()
```

Fig. 13: Exemple de model

Naturalment Django^[13] compta amb més mòduls i funcions, però he decidit explicar aquestes quatre ja que són les que més importància tenen al projecte. Més endavant, al **capítol 5**, explicaré tot més detalladament.

4 - Anàlisi i disseny

4.1 - Requeriments

Per poder utilitzar al màxim l'aplicació web, l'usuari necessita un navegador web que estigui actualitzat per poder fer servir les versions dels llenguatges mencionats al **capítol 3**, HTML 5^[6] i CSS 3^[8]. Els navegadors que suporten aquestes versions amb les seves funcions introduïdes en elles són^[15]:

- **Google Chrome**, qualsevol versió
- **Mozilla Firefox**, a partir de la versió 5
- **Opera**, a partir de la versió 12
- **Safari**, a partir de la versió 5
- **Internet Explorer**, a partir de la versió 9

4.2 - Casos d'ús

A continuació es detallaran els diferents casos d'ús que podran fer els usuaris de la pàgina web. Primer mostraré un diagrama amb tots els casos d'ús i seguidament detallaré cadascun de manera textual.

4.2.1 - Diagrama de casos d'ús

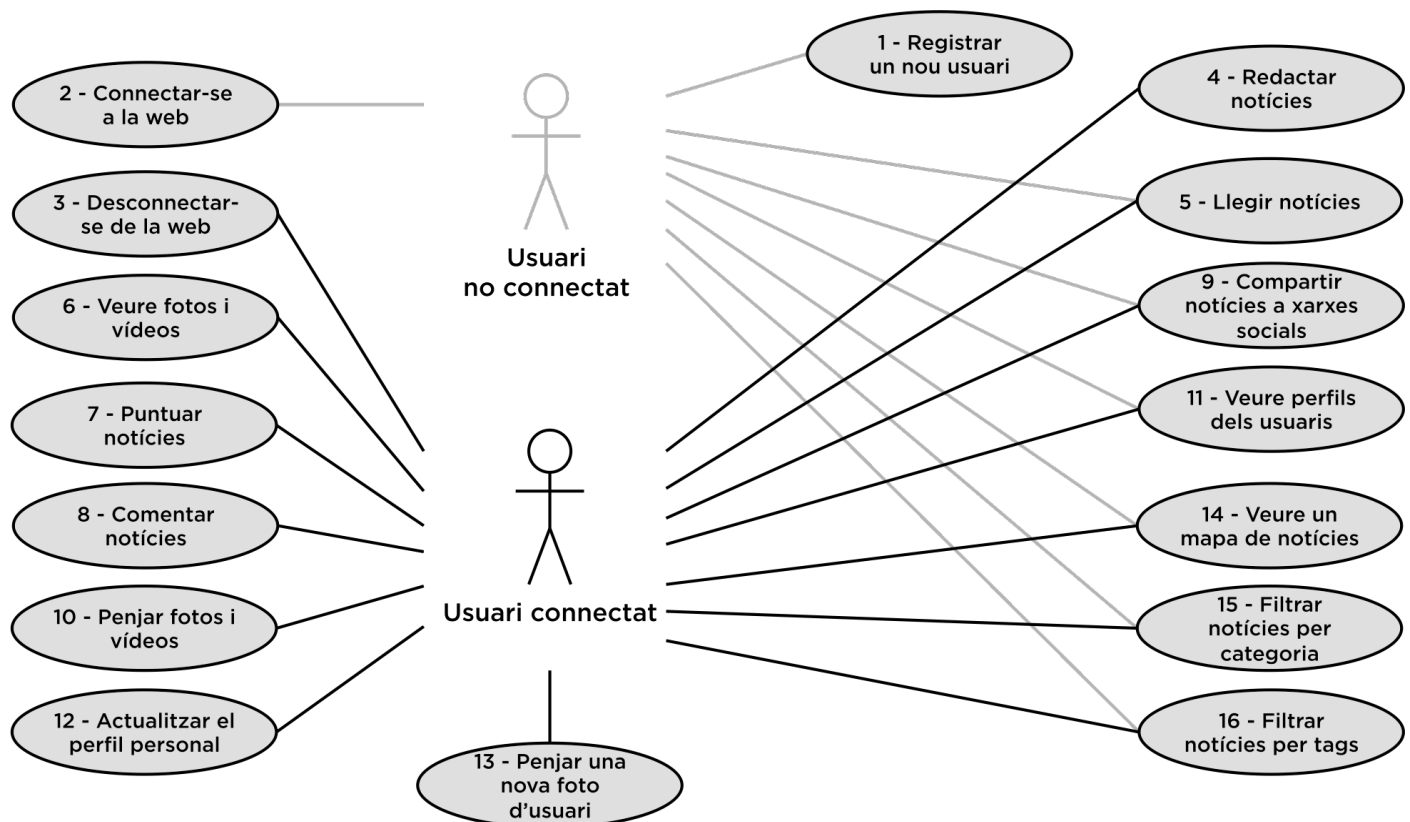


Fig. 14: Diagrama de casos d'ús en UML (en dos colors indicant els dos actors diferents)

4.2.2 - Descripció de casos d'ús textuais

1 - Registrar un nou usuari

Actors: Usuari no connectat.

Descripció: Un usuari que encara no estigui registrar a la web podrà registrar un nou usuari per a poder gaudir de totes les funcionalitats que ofereix l'aplicació i podrà tenir a més un perfil d'usuari propi on es veuran les seves notícies, fotos, vídeos...

Precondició: -

Flux bàsic:

1. L'usuari entra a la pàgina de registrar un nou usuari a través d'un link.
2. L'usuari entra les dades requerides a un formulari.
3. Si les dades són correctes es crea un nou registre a la base de dades i es crea un nou perfil d'usuari.

Flux alternatiu:

3. Si les dades no són correctes, l'aplicació redirecciona a l'usuari un altre cop a la pàgina de registre d'usuari indicant-li quines dades no són correctes i es torna al punt 2.

Postcondició: L'usuari estarà registrat a la web.

2 - Connectar-se a la web

Actors: Usuari no connectat.

Descripció: Un usuari que prèviament hagi registrat un compte nou i tingui un perfil d'usuari es podrà connectar a la web mitjançant un nom d'usuari i una contrasenya.

Precondició: -

Flux bàsic:

1. L'usuari entra les dades requerides a un formulari de *login* que està a la pàgina inicial.
2. Si les dades són correctes es connecta a l'usuari a l'aplicació.

Flux alternatiu:

2. Si les dades no són correctes, l'aplicació redirecciona a l'usuari un altre cop a la pàgina de connectar-se o pàgina inicial indicant-li quines dades no són correctes i es torna al punt 1.

Postcondició: L'usuari es connecta a l'aplicació. L'actor passa a ser "Usuari connectat".

3 - Desconnectar-se de la web

Actors: Usuari connectat.

Descripció: Un usuari que estigui connectat a l'aplicació podrà desconnectar-se de la mateixa i tancar la sessió.

Precondició: -

Flux bàsic:

1. L'usuari fa clic sobre el link de desconnectar-se.

Flux alternatiu: -

Postcondició: L'usuari es desconnecta de l'aplicació. L'actor passa a ser "Usuari no connectat".

4 - Redactar notícies

Actors: Usuari connectat.

Descripció: Un usuari connectat podrà redactar una nova notícia omplint els camps que se li indicaran en un formulari: títol, contingut i categoria com a mínim. Podran haver camps addicionals optatius.

Precondició: -

Flux bàsic:

1. L'usuari entra al link de redactar una nova notícia.
2. L'usuari entra les dades requerides al formulari de redactar una nova notícia.
3. Es crea una nova notícia a la base de dades.
4. L'usuari es redirigit a la notícia que ha estat redactada.

Postcondició: L'aplicació tindrà una notícia més a la base dades.

5 - Llegir notícies

Actors: Usuari no connectat, usuari connectat.

Descripció: Un usuari podrà seleccionar una notícia i podrà llegir el seu contingut.

Precondició: L'usuari ha de seleccionar la notícia a llegir. Al menys hi ha d'haver una notícia a l'aplicació.

Flux bàsic:

1. L'usuari selecciona la notícia que vol llegir.
2. L'usuari es redirigit a la notícia seleccionada.

Postcondició: L'usuari podrà fer accions relacionades amb la notícia (veure altres casos d'ús).

6 - Veure fotos i vídeos

Actors: Usuari no connectat, usuari connectat.

Descripció: Un usuari podrà seleccionar una fotografia o un vídeo en una notícia per veure-ho.

Precondició: L'usuari ha d'haver seleccionat una notícia.

Flux bàsic:

1. L'usuari selecciona la imatge o el vídeo que vol veure.
2. La imatge seleccionada s'amplia per veure-la en gran o el vídeo s'amplia per començar a reproduir-se.

Postcondició: -

7 - Puntuar notícies

Actors: Usuari connectat.

Descripció: Un usuari podrà donar una puntuació de l'1 al 5 a una notícia seleccionada.

Precondició: L'usuari ha d'haver seleccionat una notícia.

Flux bàsic:

1. L'usuari selecciona la puntuació que li vol donar a la notícia.
2. Es calcula la puntuació nova de la notícia gràcies a les puntuacions anteriors fent-se la mitjana.

Postcondició: La notícia passa a tenir una nova puntuació i un vot més.

8 - Comentar notícies

Actors: Usuari connectat.

Descripció: Un usuari connectat podrà escriure un comentari a una notícia omplint els camps que se li indicaran en un formulari: contingut com a mínim.

Precondició: L'usuari ha d'haver seleccionat una notícia.

Flux bàsic:

1. L'usuari entra les dades requerides al formulari de comentar.
2. Es crea un nou comentari associat a la notícia a la base de dades.
3. L'usuari es redirigit a la notícia que ha estat comentada i pot veure el seu comentari.

Postcondició: L'aplicació tindrà un comentari més a la base dades.

9 - Compartir notícies a xarxes socials

Actors: Usuari no connectat, usuari connectat.

Descripció: Qualsevol usuari podrà compartir una notícia seleccionada a les xarxes socials que estiguin disponibles, en un primer moment: Facebook, Twitter, Google+, Tumblr i Pinterest.

Precondició: L'usuari ha d'haver seleccionat una notícia.

Flux bàsic:

1. L'usuari fa clic al botó corresponent a la xarxa social a la qual vol compartir la notícia.
2. L'usuari segueix les instruccions indicades a cada xarxa social.
3. La notícia queda compartida a la xarxa social desitjada.

Postcondició: -

10 - Penjar fotos i vídeos

Actors: Usuari connectat.

Descripció: Un usuari connectat podrà, a la vegada que escriu una nova notícia, penjar fotos i vídeos a la pàgina vinculant aquests a la notícia que es crearà a la vegada. En un futur, es podrien penjar notícies i vídeos sense necessitat de crear una nova notícia.

Precondició: L'usuari ha d'estar a la pàgina de redactar una nova notícia.

Flux bàsic:

1. L'usuari selecciona una imatge o vídeo del seu sistema o bé l'arrossega a l'àrea habilitada a tal efecte.
2. La foto o vídeo es penja automàticament a la xarxa i es vincula a la notícia que s'està creant.

Postcondició: El sistema compta amb un nou arxiu multimèdia vinculat a una notícia.

11 - Veure perfils dels usuaris

Actors: Usuari no connectat, usuari connectat.

Descripció: Al perfil personal s'hi podran veure totes les notícies redactades per l'usuari, tots els arxius penjats i el número de comentaris que ha fet.

Precondició: L'usuari s'ha d'haver registrat.

Flux bàsic:

1. L'usuari entra el perfil de l'usuari que vol veure.
2. L'usuari veu les notícies redactades per l'usuari, els arxius penjats i els seus comentaris.

Postcondició: -

12 - Actualitzar el perfil personal

Actors: Usuari connectat.

Descripció: Una vegada un usuari connectat hagi entrat al seu propi perfil, tindrà la opció de canviar certes dades personals i actualitzar-les: correu electrònic, biografia, localització...

Precondició: L'usuari connectat ha d'entrar al seu propi perfil.

Flux bàsic:

1. L'usuari selecciona l'opció de modificar les dades personals.
2. L'usuari actualitza les dades que desitgi canviar.
3. L'usuari es redirigit al seu propi perfil amb les dades actualitzades.

Postcondició: Les dades de l'usuari passen a estar actualitzades.

13 - Penjar una nova foto d'usuari

Actors: Usuari connectat.

Descripció: De la mateixa forma que l'usuari pot actualitzar les seves dades personals, també podrà penjar una nova imatge que li identificarà al llarg de tota l'aplicació.

Precondició: L'usuari connectat ha d'entrar al seu propi perfil.

Flux bàsic:

1. L'usuari selecciona l'opció de modificar la imatge personal.
2. L'usuari penja una nova imatge gràcies al formulari corresponent: podrà seleccionar una imatge del seu sistema o bé arrossegar-la a l'àrea habilitada a tal efecte.
3. La foto de perfil es penja automàticament a la base de dades.

Postcondició: La imatge de perfil de l'usuari passa a estar actualitzada.

14 - Veure un mapa de notícies

Actors: Usuari no connectat, usuari connectat.

Descripció: Una de les característiques més destacables de l'aplicació web es poder visualitzar en un mapa del planeta diferents punts indicant quins són els llocs del món més actius amb les notícies més noves. Aquest mapa pot mostrar només les notícies d'una categoria en concret o pot mostra-les totes.

Precondició: -

Flux bàsic:

1. L'usuari selecciona el mapa a la pàgina principal.
2. L'usuari pot visualitzar les dades del mapa i fer clic als diferents punts del mapa per veure la notícia amb més detall.

Postcondició: -

15 - Filtrar notícies per categoria

Actors: Usuari no connectat, usuari connectat.

Descripció: Una opció per visualitzar notícies es filtrant només les d'una categoria en concret.

Precondició: Ha d'haver al menys una notícia per la categoria desitjada.

Flux bàsic:

1. L'usuari selecciona la categoria que vol.
2. L'usuari veu una pàgina on només hi apareixen les notícies de la categoria seleccionada.

Postcondició: -

16 - Filtrar notícies per tags

Actors: Usuari no connectat, usuari connectat.

Descripció: Una opció per visualitzar notícies es filtrant només les que portin uns tags concrets.

Precondició: Ha d'haver al menys una notícia pels tags seleccionats.

Flux bàsic:

1. L'usuari entra en un formulari els tags que vol buscar.
2. L'usuari veu una pàgina molt similar a la pàgina inicial però on només hi apareixen les notícies que portin els tags introduïts al formulari.

Postcondició: -

4.3 - Disseny de la base de dades

A continuació es mostrarà el diagrama del model ER de la base de dades i es detallarà concretament que conté cada taula i com són els seus atributs.

4.3.1 - Model ER

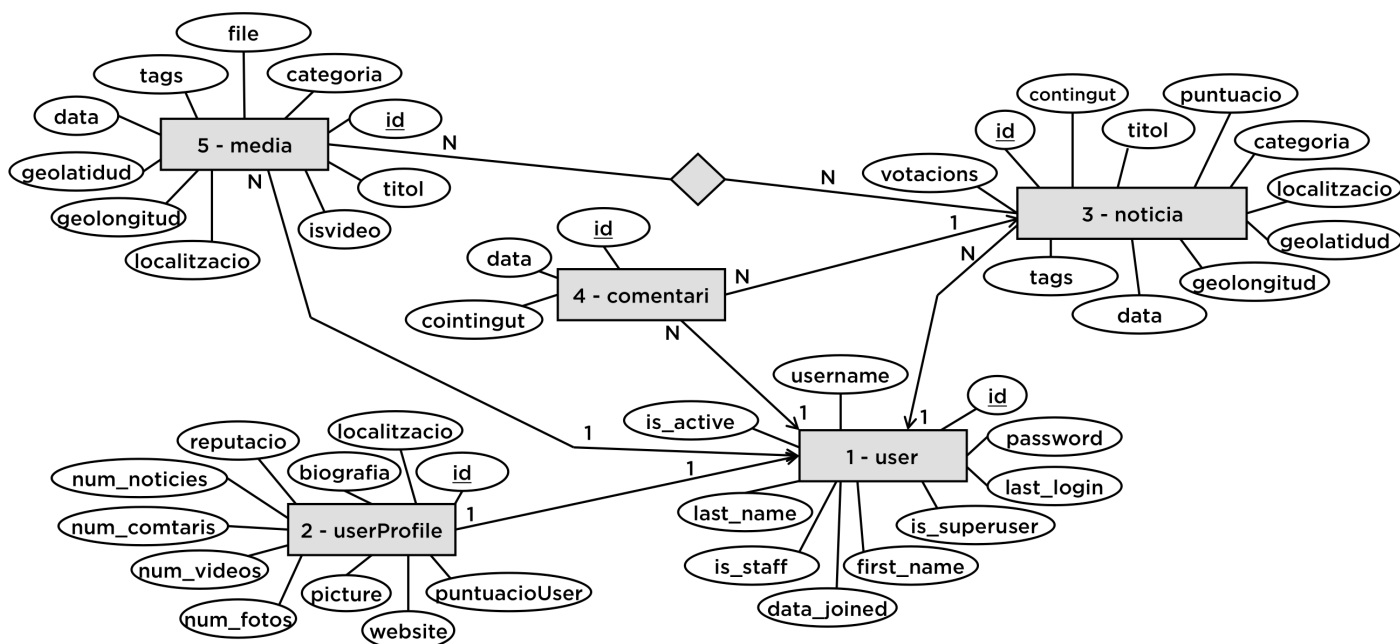


Fig. 15: Model ER de les taules utilitzades a la base de dades pel projecte

Tot i que la base de dades conté moltes més taules, a l'aplicació només es fan servir 5 de manera directa. Són les taules referents als usuaris registrats, els perfils d'usuaris, les notícies, els comentaris, i els arxius multimèdia penjats.

4.3.2 - Descripció de taules

Taula 1 - user

Aquesta és una de les taules que ja venen creades per Django^[13] de manera predefinida. Els seus atributs són aquells que defineixen la configuració bàsica de cada usuari.

id (INT (11)): clau primària i identificador únic de cada usuari.

username (VARCHAR (30)): nom d'usuari utilitzat per identificar de manera clara a l'usuari i utilitzat per poder-se connectar.

first-name (VARCHAR (30)): nom real de l'usuari, opcional.

last-name (VARCHAR (30)): cognom de l'usuari, opcional.

email (VARCHAR (75)): correu electrònic de l'usuari, opcional.

password (VARCHAR (128)): contrasenya xifrada que es fa servir per connectar l'usuari a l'aplicació.

is_staff (TINYINT (1)): paràmetre per indicar si l'usuari pot entrar al portal d'administració.

is_superuser (TINYINT (1)): paràmetre per indicar que l'usuari té tots els permisos, inclús els de staff.

last_login (DATETIME): data de la última connexió de l'usuari.

data_joined (DATETIME): data de quan l'usuari es va registrar a la web.

Taula 2 - userProfile

Aquesta taula ha estat definida pels dos projectes per tal de poder oferir als usuaris l'oportunitat de tenir un perfil propi a la pàgina web. Aquesta taula conté, a més, diferents camps que estan preparats per fer un possible futur sistema de reputacions d'usuaris que en l'actualitat no s'utilitzen.

id (INT (11)): clau primària i identificador únic de cada perfil d'usuari.

user_id (INT (11)): l'usuari al qual va associat el perfil, clau externa.

biografia (LONGTEXT): una petita descripció de l'usuari, opcional.

localitzacio (LONGTEXT): la localització actual de l'usuari, opcional.

website (VARCHAR (200)): pàgina web personal de l'usuari, opcional.

picture (VARCHAR (100)): imatge de perfil de l'usuari, opcional.

reputacio (INT (11)): reputació que té l'usuari a la web donada pel propi sistema, no utilitzat de moment.

puntuacio_user (INT (11)): la puntuació donada pels altres usuaris a l'usuari, no utilitzat de moment.

num_fotos (INT (11)): número de fotos que ha penjat l'usuari.

num_comentaris (INT (11)): número de comentaris que ha escrit l'usuari.

num_videos (INT (11)): número de vídeos que ha penjat l'usuari.

num_notícies (INT (11)): número de notícies que ha redactat l'usuari.

Taula 3 - noticia

Aquesta taula ha estat definida pels dos projectes per tal de poder penjar les notícies que estiguin redactes pels usuaris.

id (INT (11)): clau primària i identificador únic de cada noticia.

autor_id (INT (11)): l'usuari que va escriure la noticia, clau externa.

data (DATETIME): data i hora en la que es va crear la noticia.

títol (LONGTEXT): títol de la noticia.

contingut (LONGTEXT): cos de la noticia.

tags (LONGTEXT): tags associats a la noticia, opcional.

categoria (LONGTEXT): categoria de la noticia.

localitzacio (LONGTEXT): nom del lloc on va ocórrer la noticia, opcional.

geolatitud (DOUBLE): coordenada de latitud del lloc on va ocórrer la noticia, guardada per poder mostrar el punt al mapa, opcional.

geolongitud (DOUBLE): coordenada de longitud del lloc on va ocórrer la noticia, guardada per poder mostrar el punt al mapa, opcional.

puntuacio (DOUBLE): puntuació mitjana que té la noticia, per defecte és 0.

votacions (INT (11)): número de puntuacions donades a la noticia, guardat per poder calcular la mitjana, per defecte és 0.

Taula 4 - comentari

Aquesta taula ha estat definida pels dos projectes per tal de poder guardar els diferents comentaris que els usuaris facin a les notícies.

id (INT (11)): clau primària i identificador únic de cada comentari.

autor_id (INT (11)): l'usuari que va escriure el comentari, clau externa.

noticia_id (INT (11)): noticia a la que va associada el comentari, clau externa.

data (DATETIME): data i hora en la que es va crear el comentari.

contingut (LONGTEXT): cos del comentari.

Taula 5 - media

Aquesta taula ha estat definida pels dos projectes per tal de poder guardar informació sobre les imatges i vídeos que els usuaris pengin a l'aplicació.

Aquesta taula conté, a més, diferents camps que estan preparats per fer un possible futur sistema de galeries, on podrien haver imatges i vídeos que no estiguin associats a cap noticia, que en l'actualitat no s'utilitzen.

id (INT (11)): clau primària i identificador únic de cada imatge o vídeo.

autor_id (INT (11)): l'usuari que va penjar la imatge o vídeo, clau externa.

file (VARCHAR (100)): URL on està guardada físicament la imatge o vídeo.

data (DATETIME): data i hora en la que es va penjar la imatge o vídeo.

títol (LONGTEXT): títol de la imatge o vídeo, no utilitat de moment.

tags (LONGTEXT): tags associats a la imatge o vídeo, no utilitat de moment.

categoria (LONGTEXT): categoria de la imatge o vídeo, no utilitat de moment.

localitzacio (LONGTEXT): nom del lloc d'on és la imatge o vídeo, no utilitat de moment.

geolatitud (DOUBLE): coordenada de latitud del lloc d'on és la imatge o vídeo, no utilitat de moment.

geolongitud (DOUBLE): coordenada de longitud del lloc d'on és la imatge o vídeo, no utilitat de moment.

isvideo (TINYINT (1)): variable per indicar si es tracta d'un vídeo o d'una imatge.

5 - Desenvolupament

En aquest capítol explicaré el procés de desenvolupament del projecte.

5.1 - Entorns de desenvolupament utilitzats

Durant el desenvolupament del projecte he fet servir dues eines per a poder-lo dur a terme. La primera es el servidor OpenShift^[16], i la segona és el programa de repositoris GitHub^[17].

5.1.1 - OpenShift

Per tal de poder desenvolupar l'aplicació, els projectes vam decidir que necessitàvem algun servidor per poder fer la comunicació més fàcil entre la pàgina web i l'aplicació per a Android. Vam estar investigant diversos servidors i alguns d'ells els vam provar per verificar si era el que necessitàvem.

Com en un principi la part web estaria feta amb PHP^[5] només ens vam preocupar de buscar un servidor que acceptés aquest llenguatge de programació i que ens pogués proporcionar accés a bases de dades, la característica principal i que ens obligava a tenir un servidor web.

Al canviar a desenvolupar sobre Django^[13] vam haver d'instal·lar tant Python^[11] com aquest framework al servidor, però per fer-ho necessitàvem accedir a la consola del servidor i aquell en concret no ens ho permetria. En aquell moment vam començar a buscar un nou servidor que ens permetés l'accés a la consola per tal d'instal·lar Django.

Després de fer cerques durant uns quants dies vam trobar al servidor que ens oferia **OpenShift**^[16], un servidor molt diferent als que havíem estat provant, ja que aquest estava preparat específicament per a fer aplicacions web a través de diferents frameworks, i ens oferia la possibilitat d'instal·lar de manera automàtica Django. Així doncs aquest va ser el servidor escollit.

Una altra característica que té aquest servidor és que està preparat per fer proves, i per tant és gratuït i no té gaires restriccions.

5.1.2 - GitHub

Des d'un principi ambdós havíem decidit que compartiríem sempre la part de codi que fos necessària a través de Git. D'aquesta manera sempre podríem accedir als canvis que havia fet l'altre en la part comuna, quedant sense compartir la part individual per no fer-nos cap embolic a l'hora de programar.

En aquest sentit, es va decidir obrir un repositori privat a **GitHub**^[17], però va resultar que els repositoris privats eren de pagament, llavors vam haver de buscar una altre alternativa.

Va ser llavors, quan gràcies a trobar el nou servidor *OpenShift*^[16] ens va aparèixer la possibilitat de retornar a la idea inicial de compartir codi. Resulta que la manera de funcionar que té

OpenShift és mitjançant repositoris que es penjen al servidor i després es fan accessibles a la web.

Ja teníem el problema solucionat, no només volíem treballar amb repositoris de manera voluntària sinó que el propi servidor ens obligava a fer-ho. Aquest va ser un altre punt decisiu per escollir *OpenShift*^[6] com a servidor per fer el desenvolupament.

5.2 - Programació

Els arxius que apareixen al projecte es poden separar segons el patró de disseny Model-Vista-Controlador.

Tant el **Model** com el **Controlador** està format per arxius Python^[11], mentre que la **Vista** està formada per plantilles HTML^[6] i arxius estàtics (CSS^[8] i JavaScript^[9]).

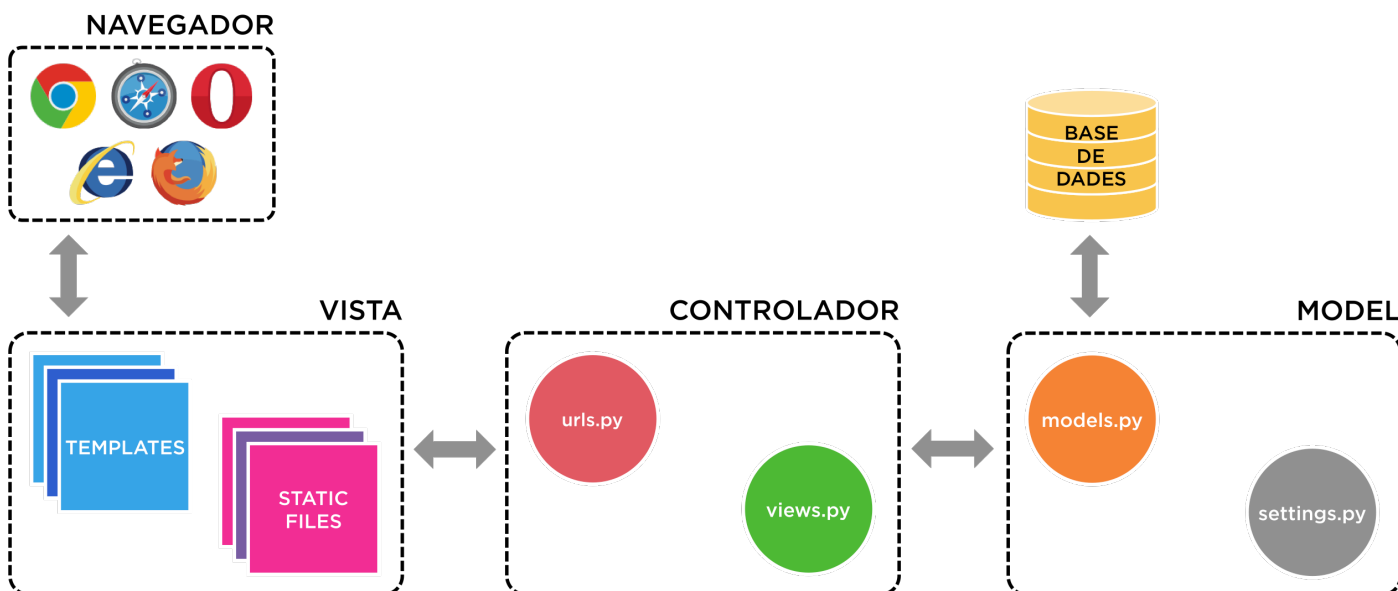


Fig. 16: Diagrama dels arxius creats al projecte i la seva relació Model-Vista-Controlador

5.2.1 - Arxius Python (Controlador i Model)

Django ens obliga a fer servir diferents arxius Python^[11] per tal de fer funcionar tot el projecte. Tot seguit passaré a explicar com funcionen.

models.py

Aquest arxiu forma part del Model de l'aplicació i ja ha estat explicat de manera general al capítol 3. El nostre en concret està definit per tal de crear les taules esmentades al capítol 4 i permetre la connexió amb la base de dades de manera fàcil i sense haver d'escriure les consultes directament en SQL.

settings.py

Aquest arxiu forma part del Model de l'aplicació i és l'arxiu de configuració del servidor de Django^[13], nosaltres no hem hagut de tocar gaire l'arxiu, només hem introduït les dades de la base de dades i dades bàsiques que eren indicades pel nostre servidor, *OpenShift*^[16].

urls.py

Aquest arxiu forma part del Controlador de l'aplicació i la seva funció és la de crear les URLs que tindrà la pàgina web. El nostre en concret conté les URLs que fem servir tant a la web com a l'aplicació per a Android. Les necessàries per aquest projecte, a la part web, són:

/

Pàgina d'inici, envia a la funció home.

admin/

Pàgina del portal d'administració (creada per Django).

checkUsername/

Pàgina per veure si un usuari existeix, associada al cas d'ús 1, envia a la funció checkUsername.

comment/

Pàgina per comentar una notícia, associada al cas d'ús 8, envia a la funció comment.

compose/

Pàgina per redactar una notícia nova, associada al cas d'ús 4, envia a la funció compose.

editprofile/

Pàgina per editar les dades d'usuari, associada al cas d'ús 12, envia a la funció *editprofile*.

error404/

Error 404: La pàgina no existeix, aquesta és la pàgina que apareix quan un usuari intenta accedir a una pàgina que no existeix, envia a la funció error404.

file/

Pàgina per penjar arxius, associada al cas d'ús 10, envia a la funció uploadfile.

fileProfile/

Pàgina per penjar una nova imatge d'usuari, associada al cas d'ús 13, envia a la funció fileProfile.

lastMedia/

Pàgina per trobar la última foto que ha penjat un usuari, associada als casos d'ús 4 i 10, envia a la funció lastMedia.

login/

Pàgina per connectar a un usuari, associada al cas d'ús 2, envia a la funció loginPage.

logout/

Pàgina per desconnectar a un usuari, associada al cas d'ús 3, envia a la funció logoutPage.

map/

Pàgina del mapa, associada al cas d'ús 14, envia a la funció `mapa`.

news/category/

Pàgina per veure les notícies d'una categoria concreta, associada al cas d'ús 15, envia a la funció `category`.

profile/

Pàgina del perfil de l'usuari actiu, associada als casos d'ús 11, 12 i 13, envia a la funció `myprofile`.

profile/username/

Pàgina del perfil d'un usuari concret, associada al cas d'ús 11, envia a la funció `profile`.

read/newsid/

Pàgina d'una notícia concreta, associada als casos d'ús 5, 6 i 9, envia a la funció `read`.

register/

Pàgina per registrar-se a la web, associada al cas d'ús 1, envia a la funció `register`.

search/

Pàgina per filtrar per tags, associada al cas d'ús 16, envia a la funció `search`.

vote/

Pàgina per puntuar notícies, associada al cas d'ús 7, envia a la funció `vote`.

views.py

Aquest arxiu, tot i el seu nom, forma part del Controlador de l'aplicació i la seva funció és la de crear el contingut que després serà mostrat a les diferents pàgines HTML, o templates. El nostre en concret conté les funcions indicades a l'arxiu `urls`. Les necessàries per aquest projecte, a la part web, són:

`home`

Redirecciona a la pàgina inicial carregant totes les notícies que hi ha a la base de dades.

`checkUsername`

Comprova si existeix l'usuari especificat. Aquesta funció és una funció auxiliar de la pàgina de registrar usuari, i serveix per indicar-li a un nou usuari si el nom que ha introduït està disponible. Aquesta funció està associada al cas d'ús 1.

`comment`

Aquesta funció rep les dades que han estat enviades pel formulari de redactar un nou comentari i crea el comentari a la base de dades. Després redirecciona a l'usuari a la notícia que ha estat comentada. Aquesta funció està associada al cas d'ús 8.

compose

Aquesta funció rep les dades que han estat enviades pel formulari de redactar una nova notícia i crea la notícia a la base de dades. Després redirecciona a l'usuari a la notícia que ha estat creada. Aquesta funció està associada al cas d'ús 4.

editprofile

Aquesta funció recull les noves dades que l'usuari entra al formulari d'actualitzar el perfil i les actualitza a la base de dades. Aquesta funció està associada al cas d'ús 12.

error404

Aquesta funció serveix per redirigir a un usuari que ha buscat una pàgina que no existeix o no està definida al arxiu URLs a una pàgina concreta que fa substitueix la pàgina d'error 404 convencional.

uploadfile

Aquesta funció es crida automàticament pel formulari de penjar fotos o vídeos i crea un registre a la base de dades amb les dades de la imatge o vídeo penjada. Aquesta funció està associada al cas d'ús 10.

fileProfile

Aquesta funció es crida automàticament pel formulari d'actualitzar la foto d'usuari i actualitza el registre a la base de dades amb les dades de la nova imatge. Aquesta funció està associada al cas d'ús 13.

lastMedia

Funció que serveix per dir al sistema quina ha estat la última imatge o vídeo que ha penjat un usuari. És una funció auxiliar que utilitza el formulari de redactar una nova notícia per tal de vincular les imatges o vídeos a la notícia. Aquesta funció està associada als casos d'ús 4 i 10.

loginPage

Funció que comprova si l'usuari entrar existeix a la base de dades i està actiu i el connecta a l'aplicació. Aquesta funció està associada al cas d'ús 2.

logoutPage

Funció que desconnecta a un usuari de la pàgina. Aquesta funció està associada al cas d'ús 3.

mapa

Funció que redirecciona a l'usuari a la plantilla del mapa, explicada al següent apartat. Aquesta funció està associada al cas d'ús 14.

category

Funció que funciona de manera molt similar a *home* però que en comptes d'agafar totes les notícies només retorna aquelles que siguin d'una categoria en concret. Aquesta funció està associada al cas d'ús 15.

myprofile

Funció que redirecciona a l'usuari al seu perfil personal. Està feta per separat de la següent funció degut a que aquesta no necessita cap paràmetre mentre que l'altre necessita indicar-li el perfil a visitar. Aquesta funció està associada al cas d'ús 11.

profile

Funció que funciona igual que *myprofile* però redireccionant al perfil indicat per un paràmetre. Aquesta funció està associada al cas d'ús 11.

read

Funció que carrega de la base de dades una notícia indicada i redirecciona a l'usuari a una plantilla on es pot llegir aquesta notícia i veure les fotos i vídeos relacionades. Aquesta funció està associada als casos d'ús 5 i 9.

register

Funció que recull les dades introduïdes pel formulari de registrar a un usuari i el crea a la base de dades. També crea addicionalment un nou perfil per l'usuari recentment creat. Després el connecta a l'aplicació i el redirecciona al seu perfil. Aquesta funció està associada al cas d'ús 1.

search

Aquesta funció s'encarrega d'obtenir les dades entrades al buscador que hi ha sempre a la part superior de la pàgina web i mostra les notícies que continguin els tags especificats al buscador. Aquesta funció està associada al cas d'ús 16.

vote

Aquesta funció s'encarrega de calcular la nova puntuació que tindrà una notícia i l'introdueix a la base de dades. La funció ha de rebre la notícia a la qual calcular la nova puntuació i els punts que li ha donat un usuari concret. Aquesta funció està associada al cas d'ús 7.

mailuser

Funció que envia un missatge de correu electrònic a un usuari. Aquesta funció no es utilitzada de moment, però ha estat creada per poder, en un futur, fer que el registre d'usuaris necessiti verificar per correu electrònic el registre i així evitar spam. Aquesta funcionalitat no es pot fer de moment.

5.2.2 - Plantilles HTML (Vista)

Com ja he explicat abans, les plantilles, o *templates*, serveixen per mostrar a l'usuari les pàgines que veurà al navegador. Gràcies a les funcions indicades a l'arxiu `views.py`, aquestes poden mostrar contingut dinàmic calculat prèviament.

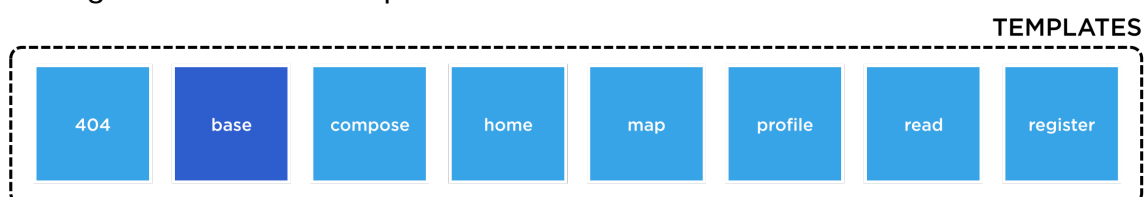


Fig. 17: Diagrama de les plantilles HTML que apareixen al projecte, totes hereten contingut de base

A continuació explico el seu contingut:

404

Aquesta és la pàgina que els usuaris veuran quan intentin accedir a una pàgina que no existeix. Està creada per tal de mostrar una pàgina molt més visual i més clarificadora que la típica pàgina d'error 404 que està definida pel propi navegador.



Fig. 18: Pàgina d'error 404

Base

Aquesta plantilla conté dues parts: per un lloc la barra superior de la web amb els diferents links a les categories, el mapa, el filtrador per tags, etc. I per altra banda té el footer on es mostra el copyright i els avisos legals de l'aplicació.

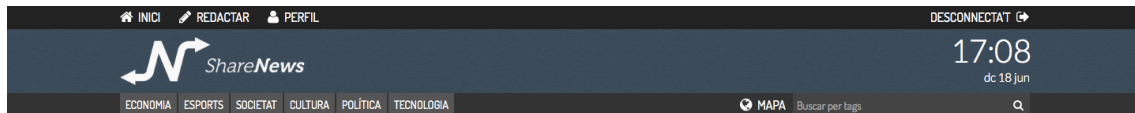


Fig. 19: Pàgina "base"

Es tracta de dues parts que s'han de repetir a totes les pàgines de la web, i per això estan en un arxiu separat, per poder després importar-les a les demès plantilles sense haver de copiar el codi a les demès amb el greuge de que si modifiquéssim qualsevol detall hauríem d'anar una per una modificant sempre la mateixa línia de codi. Conté també les importacions de llibreries externes^[18].

A més a més, conté un link que permet a l'usuari fer el cas d'ús 3, un buscador per poder fer el cas d'ús 16, i les categories per fer el cas d'ús 15.

Compose

Aquesta plantilla permet a l'usuari realitzar els casos d'ús 4 i 10. Aquesta pàgina li mostra a l'usuari un formulari on poder entrar les dades de la notícia que vol crear. Aquest formulari compta amb els camps obligatoris de títol, contingut i categoria. A més, també permet al redactor que pugui definir els tags que compondran la notícia, ficar la localització on ha ocorregut la notícia gràcies al sistema de geolocalització que incorpora HTML 5^[6], i a més permet penjar fotos i vídeos relacionats amb la notícia.

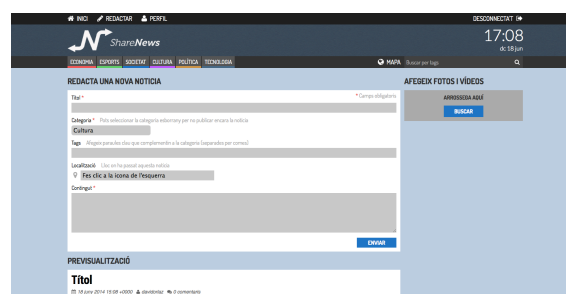


Fig. 20: Pàgina de redactar una notícia

Té la característica de que li permet a l'usuari poder veure com quedarà la notícia un cop hagi estat publicada a la web.

Home

Aquesta plantilla permet a l'usuari realitzar principalment el cas d'ús 2, i a més conté links a altres plantilles per fer altres casos d'ús.

Aquesta és la plantilla que defineix la pàgina inicial de l'aplicació. També utilitza aquesta mateixa plantilla la funció category per mostrar només les notícies d'una categoria concreta. Mostra també un minimapa on poder veure els punts més actius del planeta de manera ràpida.



Fig. 21: Pàgina d'inici

Map

Aquesta plantilla permet a l'usuari realitzar el cas d'ús 14. Aquesta és la plantilla que defineix la pàgina que els usuaris podran veure quan vulguin veure el mapa de notícies a pantalla completa.

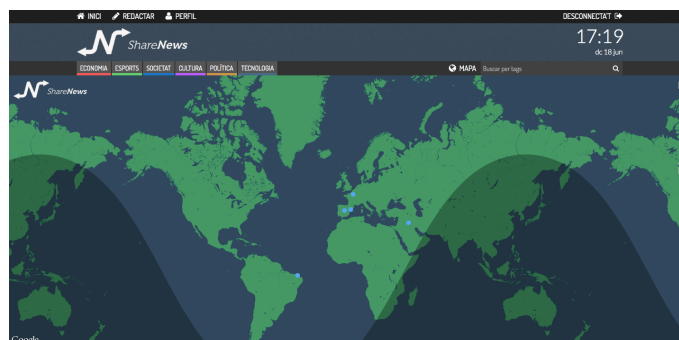


Fig. 22: Pàgina del mapa

Profile

Aquesta plantilla permet a l'usuari realitzar principalment els casos d'ús 11, 12 i 13.

Aquesta és la plantilla que defineix els diferents perfils d'usuari. Inclou dades com totes les notícies redactades per l'usuari, tots els arxius penjats i el número de comentaris que ha fet. A més també té un minimapa on poder veure els punts més actius de les notícies escrites per aquest usuari.



Fig. 23: Pàgina del perfil d'usuari

Read

Aquesta plantilla permet a l'usuari realitzar els casos d'ús 5, 6, 7, 8 i 9.

Aquesta és la plantilla que defineix l'estructura que tindran les notícies seleccionades per poder llegir. Des d'aquesta pàgina també es mostraran els comentaris que estiguin associats a les notícies, les fotos i vídeos que tinguin relació amb aquesta i també permetrà poder escriure un nou comentari. Inclourà també els botons necessaris per a poder compartir la notícia a les xarxes socials^{[19] [20] [21] [22] [23]}.



Fig. 24: Pàgina de llegir una notícia

Register

Aquesta plantilla permet a l'usuari realitzar el cas d'ús 1.

Aquesta pàgina li mostra a l'usuari un formulari on poder entrar les dades per a poder registrar un nou usuari a l'aplicació. Té una funció associada per a indicar-li si el nom d'usuari ja existeix.

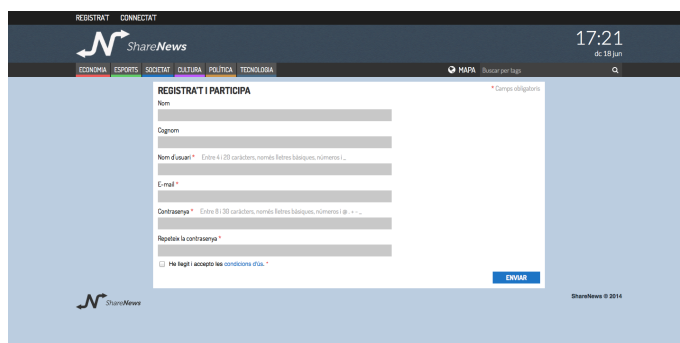


Fig. 25: Pàgina de registrar-se

6.2.3 - Arxius estàtics (Vista)

Una característica encara no mencionada a la memòria que té Django^[13] perquè no és obligatòria, però que he fet servir també, són els arxius estàtics. Aquests arxius es caracteritzen pel fet que normalment no es modifiquen amb freqüència. Són aquests els arxius que estan escrits en CSS^[8] o en JavaScript^[9].

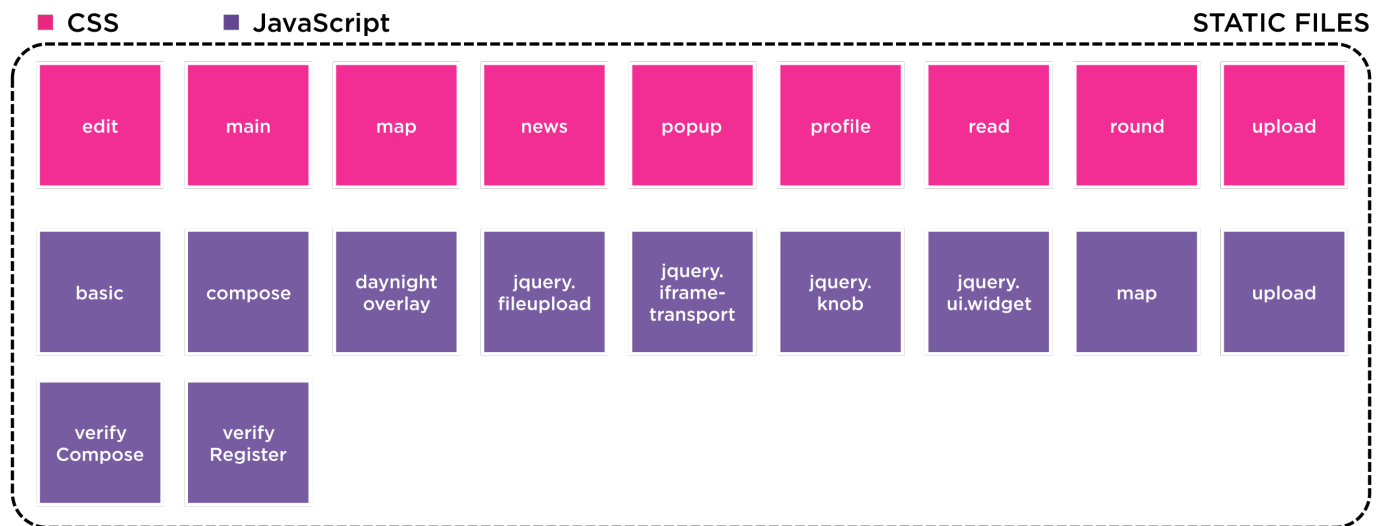


Fig. 26: Diagrama dels arxius estàtics que conté el projecte

Arxius CSS

Com ja he explicat prèviament, els arxius CSS^[8] són aquells que defineixen els estils que tindran les pàgines HTML^[6], en aquest cas, els arxius que han estat dissenyats per dotar d'estil a les plantilles esmentades prèviament són:

edit

Aquest és l'arxiu associat a la plantilla Compose.

main

Aquest és l'arxiu que dona estil a gran part de la pàgina principal, carrega tots els altres arxius CSS (ja que aquest és l'únic que carrega la plantilla base) i també conté l'estil de la plantilla Register.

map

Aquest és l'arxiu associat a la plantilla Map, dona estil als mapes que apareixen a la web.

news

Aquest és l'arxiu que dona estil a les "fitxes" de les notícies de la pàgina principal i al formulari de connectar-se a l'aplicació.

popup

Aquest és l'arxiu que dona estil als pop-ups que apareixen a l'aplicació: el de llegir les condicions de la web al registrar-se i el que s'obre per veure una imatge o un vídeo en pantalla completa.

profile

Aquest és l'arxiu associat a la plantilla Profile, dona estil als perfils dels usuaris.

read

Aquest és l'arxiu associat a la plantilla Read, dona estil a les notícies i als comentaris que tenen cadascuna.

round

Aquest és un fitxer auxiliar que permet que certs requadres de la pàgina web sigui arrodonits i no quadrats com son per defecte. És un fitxer purament estètic per no haver de repetir certes línies de CSS cada vegada que vulguem fer requadres arrodonits o rodones.

upload

Aquest és l'arxiu que dona estil al formulari i a l'àrea "Drag and Drop" que permet penjar fotos i vídeos de manera molt simple i ràpida.

Arxius JavaScript

Per tal de complementar les plantilles i donar-les una mica de moviment he creat diverses funcions JavaScript^[9] per a fer algunes funcions addicionals i fer que l'usuari se senti més còmode a la pàgina web. Tot seguit passo a explicar aquestes funcions.

Tots aquests arxius han estat creats, a menys que es tracti d'una llibreria externa, des de 0 per aquest Treball de Fi de Grau.

basic

Aquets arxiu no conté funcions com a tal, sinó que conté les configuracions que permeten fer aparèixer un rellotge a la part superior de la pàgina que s'actualitza a temps real i conté la configuració per a compartir a les xarxes socials.

compose

Aquest arxiu permet que quan un usuari estigui escrivint una nova notícia, a sota se li mostri una previsualització de com quedarà aquesta notícia al ser publicada. També s'encarrega de substituir els salts de línia introduïts al formulari en salts de línia traduïts a HTML^[6] i també conté una funció per poder ampliar o disminuir el tamany de lletra de la notícia.

daynightoverlay^[24]

Aquest arxiu és realment una llibreria externa que permet que el mapa tingui una capa que simula la llum terrestre i permet visualitzar a quins punts del planeta és de dia o de nit.

jquery.fileupload^[25]

Aquest arxiu és realment una llibreria externa que serveix per al plugin Drag & Drop per a penjar imatges o vídeos a l'hora de redactar una nova notícia.

jquery.iframe-transport^[25]

Aquest arxiu és realment una llibreria externa que serveix com a suport per al plugin Drag & Drop per a penjar imatges o vídeos a l'hora de redactar una nova notícia.

jquery.knob^[26]

Aquest arxiu és realment una llibreria externa que serveix com a suport per al plugin Drag & Drop per a penjar imatges o vídeos a l'hora de redactar una nova notícia. En concret crea els petits diagrames circulars que indiquen el percentatge del procés que té un arxiu penjat.

jquery.ui.widget^[27]

Aquest arxiu és realment una llibreria externa que serveix com a suport per al plugin Drag & Drop per a penjar imatges o vídeos a l'hora de redactar una nova notícia.

map

Aquest arxiu és el que conté totes les funcions que estan relacionades amb els mapes que es veuen a la web. Conté el mètode initialize que demana la API de Google Maps^[28] i totes les configuracions necessàries que ha de tenir per poder mostrar el mapa bé.

upload

Aquest arxiu és un arxiu que ens ajuda a connectar el plugin Drag & Drop amb la funció de penjar arxius en Python^[11]. Serveix com a pont, agafant les dades dels altres arxius ja esmentats abans i enviant-les a la funció corresponent.

verifyCompose

Aquest arxiu comprova que les dades introduïdes al formulari de redactar una nova notícia tenen el format correcte i mostra a l'usuari possibles errors. També és connecta mitjançant AJAX a la funció de recuperar l'últim arxiu que ha penjat un usuari per a guardar una llista de fotos i vídeos associats a la notícia. També conté un mètode per obtenir la geolocalització de l'usuari per a poder guardar-la a la notícia.

Quan l'usuari envia les dades, aquest arxiu comprova que siguin correctes i, si ho són, les envia a la funció corresponent a views, i si no li mostra missatges d'errors a l'usuari segons quina dada sigui la incorrecta.

verifyRegister

Aquest arxiu comprova que les dades introduïdes al formulari de registrar un nou usuari tenen el format correcte i mostra a l'usuari possibles errors. També és connecta mitjançant AJAX a la funció de comprovar si un usuari existeix a la base de dades per mostrar al client si es possible registrar aquest username o no.

Quan l'usuari envia les dades, aquest arxiu comprova que siguin correctes i, si ho són, les envia a la funció corresponent a views, i sinó li mostra missatges d'errors a l'usuari segons quina dada sigui la incorrecta.

5.3 - API de Google Maps

Per fer totes les funcions del mapa hem hagut de fer servir la API de Google Maps^[28]. Explicaré de manera breu com funciona aquesta API i que hem fet servir d'ella.

Inicialment necessitem posar-nos en contacte amb Google per a poder aconseguir una API Key que ens serveixi per a poder mostrar el mapa. Una vegada hem obtingut aquesta API Key l'hauré d'introduir mitjançant un link JavaScript^[9] a totes les pàgines on la volguem utilitzar, en el nostre cas a la plantilla base, ja que totes hereten d'aquesta.

Les configuracions que hem fet han estat bàsicament les de modificar l'aparença del mapa per a que tingués l'estil que segueix tota l'aplicació i així no desentonés, la d'introduir unes coordenades inicials que servien de centre al mapa i un zoom inicial i màxim.

Finalment la part més important del mapa era la part de mostrar els diferents marcadors. Per fer això hem creat una funció separada que creï un per un cada marcador, que simbolitzarà cada notícia, i després els agrupés a un Array que li passaríem al mapa per a que els mostrés.

Per a més informació sobre com fer servir aquesta API, es pot trobar un enllaç al seu tutorial i a la API completa al capítol de referències bibliogràfiques.

6 - Resultats i test

Després de finalitzar tot el desenvolupament vaig provar una a una cadascuna de les funcions que havien estat programades. Vaig comprovar que totes funcionaven bé, que feien el que havien de fer i que la informació que es passava entre les diferents pàgines i mètodes era la correcta. Seguidament vaig provar tots els casos d'ús per verificar que es podien completar amb èxit.

Després de fer les proves jo mateix, el següent pas era fer proves amb usuaris reals.

6.1 - Metodologia

Una vegada la pàgina web va estar acabada, vaig demanar a voluntaris que provessin algunes funcions de la web per comprovar si era fàcil d'utilitzar i veure els possibles errors que no havien estat previstos. Les dades bàsiques d'aquests participants són les següents:

	Edat i gènere	Ocupació	Coneixements informàtics	Familiarització amb xarxes socials
Usuari 1	21 / dona	Universitat	Alt	Molt alt
Usuari 2	17 / home	Batxillerat	Alt	Molt alt
Usuari 3	19 / home	Cicle Form.	Alt	Molt alt
Usuari 4	49 / dona	Mest. casa	Baix	Mitjà
Usuari 5	21 / home	Universitat	Molt alt	Molt alt
Usuari 6	23 / home	Universitat	Molt alt	Molt alt
Usuari 7	22 / dona	Universitat	Molt alt	Molt alt
Usuari 8	46 / dona	Perruquera	Baix	Molt baix
Usuari 9	20 / dona	Cicle Form.	Mitjà	Alt
Usuari 10	51 / home	Pintor	Molt baix	Molt baix

Aquests usuaris van haver de realitzar tres tasques concretes, la primera era registrar-se a la pàgina web com a nou usuari. Seguidament havien d'escriure una notícia nova sobre la temàtica que ells volguessin. Finalment, l'última activitat que havien de fer era llegir una de les notícies i comentar-la. La notícia era en concret una de la categoria d'esports, per tant també comprovaríem si sabien trobar-la. Després de fer aquestes tres activitats els participants al test van haver de contestar a una sèrie de preguntes en un post-qüestionari que es detallen a continuació:

P1 - En relació a la facilitat d'ús de l'aplicació, he trobat que és fàcil o difícil d'utilitzar:

(Molt difícil) 1 2 3 4 5 (Molt fàcil) Sense resposta

P2 - Els elements del menú estan ben organitzats i les funcions són fàcils de trobar:

(En desacord) 1 2 3 4 5 (Estic d'acord) Sense resposta

P3 - Vaig comprendre de seguida la funció de cada element del menú.

(En desacord) 1 2 3 4 5 (Estic d'acord) Sense resposta

P4 - Totes les funcions que esperava trobar en els menús eren presents.

(En desacord) 1 2 3 4 5 (Estic d'acord) Sense resposta

P5 - Trobo que la navegació per l'aplicació és:

(Molt difícil) 1 2 3 4 5 (Molt fàcil) Sense resposta

P6 - La meva impressió general de l'aplicació és:

(Negativa) 1 2 3 4 5 (Positiva) Sense resposta

Les tres primeres preguntes estan relacionades amb la facilitat d'ús de l'aplicació, la quarta està enfocada a la satisfacció que tenen els usuaris, la cinquena fa referència a la navegabilitat de l'aplicació i la última ens diu quina és la impressió general que tenen els participants després de provar l'aplicació:

6.2 - Resultats

Després de recollir les respostes dels tests hem obtingut aquests resultats:

	Registrar-se	Redactar noticia	Comentar noticia	TOTAL
Usuari 1	00:28	00:52	00:34	01:54
Usuari 2	00:49	00:48	00:27	02:04
Usuari 3	00:30	00:56	00:30	01:56
Usuari 4	02:59	03:10	01:54	08:03
Usuari 5	00:32	00:41	00:53	02:06
Usuari 6	00:27	00:45	00:23	01:35
Usuari 7	00:38	01:02	00:32	02:12
Usuari 8	03:34	04:02	02:13	09:49
Usuari 9	01:12	02:27	00:57	04:36
Usuari 10	03:57	05:19	01:49	11:05
MITJANA	01:30	02:00	01:01	04:32

Temps que han trigat en realitzar cada activitat els diferents participants mesurat en minuts i segons

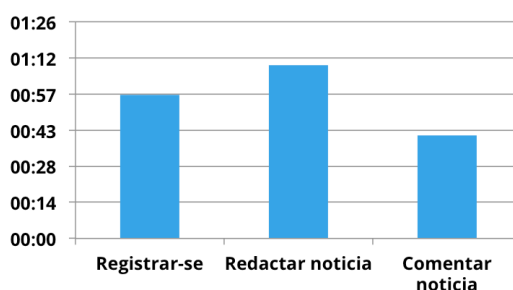


Fig. 27: Gràfic corresponent al temps mitjà que es triga en realitzar les tres activitats

Si mirem aquestes dades, podem veure que per norma general la tasca que més han trigat a fer els participants és la de redactar una nova noticia.

Això és normal, ja que és una activitat que implica haver d'escriure una notícia nova emplenant tots els camps que siguin possible, mentre que les altres dos tasques són més ràpides de fer, la de registrar-se només és emplenar uns camps que no tenen més d'una paraula cadascun, i la d'escriure un comentari és simplement escriure un text molt breu.

Analitzant un a un cada usuari ens trobem amb diferents situacions que s'expliquen a continuació:

Usuari 1

La primera usuària en fer el test és estudiant universitària, diu tenir un nivell alt de coneixements informàtics i diu estar molt familiaritzada amb les xarxes socials. El seu test va transcórrer amb bastant fluïdesa, trobant ràpidament els enllaços per a poder realitzar les activitats sense gaires dificultats.

A l'hora de buscar una notícia concreta va fer servir el buscador, però en comptes de buscar per tags va buscar per continguts, per tant va haver de corregir i tornar a la pàgina inicial per a buscar la notícia. Aquest fet és important, ja que ens indica que al buscador de tags hauria d'estar millor assenyalat que només serveix per tags i no per contingut.

Usuari 2

El segon usuari en fer el test és estudiant de batxillerat, diu tenir un nivell alt de coneixements informàtics i diu estar molt familiaritzat amb les xarxes socials. El seu test va transcórrer amb bastant fluïdesa, trobant ràpidament els enllaços per a poder realitzar les activitats sense gaires dificultats.

A l'hora de voler indicar la localització de la notícia va fer clic sobre el requadre per escriure-la manualment. Això no és possible a la pàgina web, doncs només es pot fer amb geolocalització i no de manera manual. Aquest fet és important, ja que ens indica que al requadre d'indicar la localització de la notícia hauria d'estar millor assenyalat que només es pot fer automàticament i no també manualment.

Usuari 3

El tercer usuari en fer el test és estudiant de cicle formatiu, diu tenir un nivell alt de coneixements informàtics i diu estar molt familiaritzat amb les xarxes socials. El seu test va transcórrer amb bastant fluïdesa, trobant ràpidament els enllaços per a poder realitzar les activitats sense gaires dificultats. De fet, aquest usuari no va trobar cap problema en fer el test.

Usuari 4

La quarta usuària en fer el test és una mestressa de casa, diu tenir un nivell baix de coneixements informàtics i diu estar suaument familiaritzada amb les xarxes socials. El seu test va durar més que els altres, possiblement motivat pel fet de la falta d'experiència amb les xarxes socials i dels baixos coneixements informàtics. Tot i així, tampoc no va trobar gaires problemes a l'hora de realitzar les diferents activitats demanades.

Usuari 5

El cinquè usuari en fer el test és un estudiant universitari, té un nivell molt alt en coneixements informàtics i diu estar molt familiaritzat amb les xarxes socials. El seu test va ser força ràpid, no va tenir cap inconvenient en trobar cap opció que li fes falta i va completar les tasques sense cap incident. Va destacar que la informació donada pels mapes que apareixien a la pàgina web era molt útil.

Usuari 6

El sisè usuari en fer el test és un altre estudiant universitari, té un nivell molt alt en coneixements informàtics i està molt familiaritzat amb les xarxes socials. El seu test va ser el que menys temps va durar. Va trobar cada opció de manera molt ràpida i va completar totes les tasques que se li havien demanat sense cap mena de problema. Tot i així, va dir que l'aplicació era simple.

Usuari 7

La setena usuària en fer el test és també estudiant universitària, té un nivell molt alt en coneixements informàtics i està molt familiaritzada amb les xarxes socials. De tots els estudiants universitaris, aquesta participant va ser la que més va trigar. En qualsevol cas, no va tenir cap problema important, però va destacar el fet de que una vegada que la notícia estava redactada no es podia editar.

Usuari 8

La vuitena usuària és perruquera. Els seus coneixements informàtics són baixos, i no està gens familiaritzada amb les xarxes socials. Degut als seus baixos coneixements va trigar força estona en completar les tasques, i els problemes que hi va tenir van estar simplement saber trobar on estava cada opció en cada moment.

Usuari 9

La novena usuària és estudiant de cicle formatiu. Els seus coneixements informàtics no són ni gaire baixos ni tampoc gaire alts, però està familiaritzada amb les xarxes socials. No va trigar especialment massa en fer el test, però va mostrar alguna dificultat a l'hora de saber el funcionament de cada opció que apareix a la pàgina web.

Usuari 10

L'últim usuari en fer el test és pintor. No té cap tipus de coneixements informàtics ni ha tingut cap contacte directe amb xarxes socials. El seu test és, com era d'espera, molt lent. Li costava trobar totes les opcions i entendre per a que servien, però amb paciència va completar les tasques amb èxit.

Aquest usuari és important, ja que ens informa de com usuaris que no han tingut cap mena de contacte amb altres xarxes socials es poden moure per l'aplicació i ens mostra els problemes que hi poden trobar.

En quant a les respostes del post-qüestionari, aquestes són les dades que els participants van respondre:

	P1	P2	P3	P4	P5	P6
Usuari 1	4	4	4	4	4	5
Usuari 2	4	5	5	5	5	5
Usuari 3	5	4	5	4	5	5
Usuari 4	5	5	5	5	5	5
Usuari 5	5	4	4	3	5	4
Usuari 6	4	4	4	4	5	3
Usuari 7	5	4	4	4	5	5
Usuari 8	3	3	3	4	4	5
Usuari 9	4	4	4	4	5	4
Usuari 10	3	4	3	4	4	5
MITJANA	4,2	4,1	4,1	4,1	4,7	4,6

Respostes del qüestionari

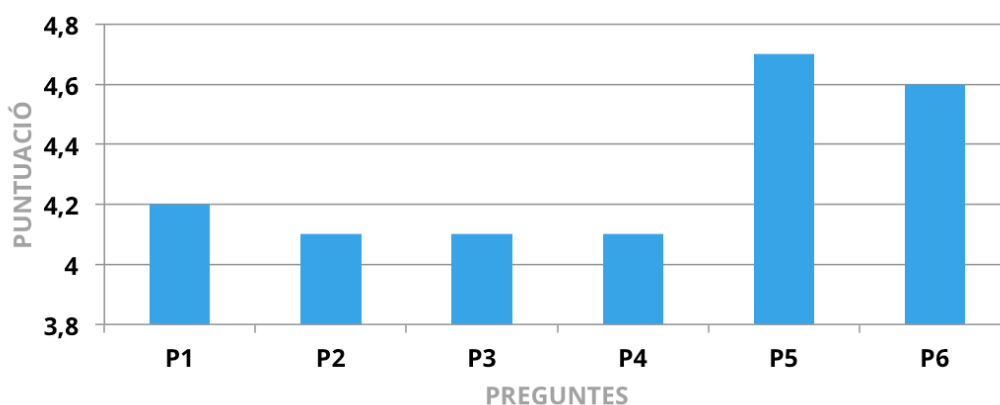


Fig. 28: Gràfic corresponent a les notes mitjanes obtingudes al qüestionari

Analitzant les estadístiques es pot veure que, en general, els usuaris voluntaris han donat molta bona puntuació a l'aplicació. Les preguntes que fan referència a la usabilitat obtenen bones notes, segons els participants l'aplicació és fàcil d'utilitzar i les funcions són fàcils de trobar.

La satisfacció en vers a l'aplicació també està ben valorada. A més, els participants creuen que la navegabilitat de la pàgina web és molt bona, donant-li molta bona puntuació a la pregunta 5, és la pregunta que està millor valorada. I per finalitzar, els usuaris donen notes molt positives a l'aplicació, tenint una bona impressió sobre la mateixa.

També puc dir que aquests tests d'usuari m'han ajudat a detectar errors que no havia detectat abans i que podrien ser corregits en el futur, com per exemple que la funció de buscar per tags és malinterpretada per alguns usuaris, ja que més d'un usuari va fallar en fer-la servir, i molts usuaris tampoc van saber per a que servia el camp de localització a l'hora de redactar notícies. Tot i això, les activitats de registrar usuaris i de comentar notícies no van suposar molts problemes pels participants.

7 - Estudi econòmic

Abans de fer l'estudi econòmic mostraré en un diagrama de Gantt com ha estat repartit el temps de dedicació d'aquest Treball de Fi de Grau. La dedicació normal ha de ser de 450 hores, i aquestes són les que jo he complert. Alguns dies feia més feina que altres, però aquestes hores de més es completaven amb dies que aprofitava per descansar o que no podia dedicar al projecte. Així doncs, el diagrama de Gantt és el següent, quedant repartit en sis fases:

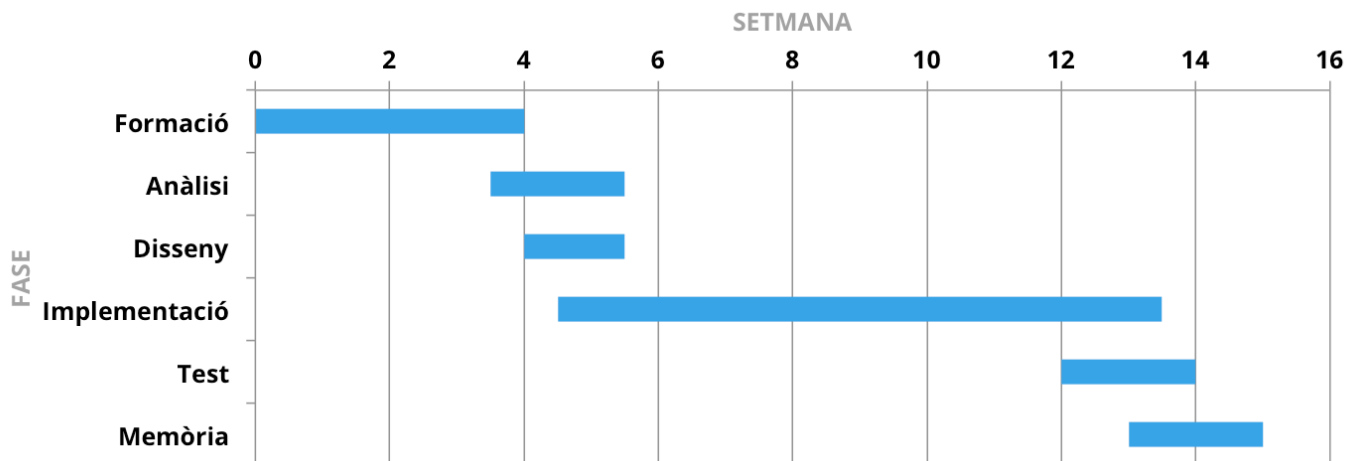


Fig. 29: Diagrama de Gantt que mostra les setmanes d'activitat de cada fase

Les sis fases són:

- **Formació:** Temps dedicat a la recerca d'informació, aprenentatge dels diferents llenguatges, aprenentatge del framework i proves inicials sobre el framework.
- **Anàlisi:** Temps dedicat als requeriments i els casos d'ús.
- **Disseny:** Temps dedicat al disseny de la base de dades i els mòduls a implementar.
- **Implementació:** Temps dedicat a la programació de tota l'aplicació amb la seva interfície gràfica.
- **Test:** Temps dedicat a fer tant les proves per part meva com als tests amb usuaris reals.
- **Memòria:** Temps dedicat a redactar la memòria del projecte.

Naturalment no totes les hores de la setmana estaven dedicades a la mateixa fase, i per tant hi havien setmanes on dues fases coincidien i anava alternant entre les dues. Finalment, i en el còmput d'hores, les fases queden repartides de la següent manera: formació 120 hores, anàlisi 30 hores, disseny 15 hores, implementació 185 hores, test 10 hores i memòria 90 hores.

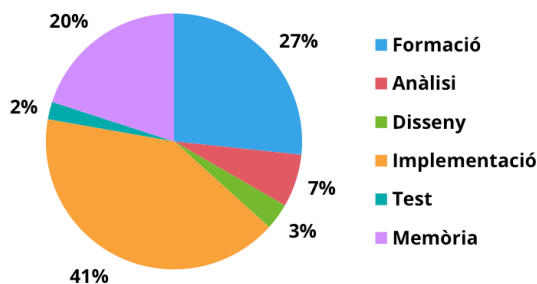


Fig. 30: Diagrama que mostra el temps proporcional dedicat a cada fase

La importància de cadascuna de les fases fixa també el seu valor econòmic i amb això tenim aquestes dades: per a la **formació** es computen 5 euros per cada hora invertida, ja que és la fase que menys importància té a l'aplicació, ja que els coneixements una vegada adquirits ja no es perden; per a l'**anàlisi** es computen 10 euros cada hora; per al **disseny** es computen 20 euros la hora, ja que és una fase de les més importants; la **implementació** és una altra de les fases més importants i on es treballa més activament, es computa amb 30 euros la hora; la fase de **test** la valorarem amb 10 euros la hora; i finalment l'elaboració de la **memòria** està valorada en 10 euros la hora també.

En resum, la valoració econòmica es pot resumir en aquesta taula:

	Hores	€/Hora	Inversió
Formació	120 h	5 €/h	600 €
Anàlisi	30 h	10 €/h	300 €
Disseny	15 h	20 €/h	300 €
Implementació	185 h	30 €/h	5.550 €
Test	10 h	10 €/h	100 €
Memòria	90 h	10 €/h	900 €
TOTAL	450 h	-	7750 €

Com que totes les tecnologies utilitzades a l'aplicació són Open Source no suposen un cost addicional. Les llicències que he fet servir també són gratuïtes.

8 - Conclusions i treball futur

Com a conclusió, després de finalitzar tot el treball i fer totes les proves pertinents podem concloure que totes les funcions i activitats planificades s'han complert satisfactòriament.

Durant el transcurs d'aquests mesos de treball i esforç he adquirit coneixements nous i les funcions planificades funcionen tal i com s'havia previst. Els objectius han estat assolits, i l'aplicació fa el que estava planificat a l'anàlisi i disseny.

A més, els usuaris reals l'han valorat de manera molt positiva i no s'han trobat amb gaires problemes a l'hora de fer les tasques que volien fer, tant si eren demanades al test com si volien provar alguna cosa addicional.

El treball ha estat molt satisfactori i s'han superat tots els reptes que han anat apareixent.

A l'actualitat la xarxa social implementada ja fa totes les funcions que havien estat planificades i ja pot ser utilitzada de manera normal sense problemes. Tot i això, aquesta aplicació encara pot rebre moltes millores.

Durant el transcurs del desenvolupament i del disseny van aparèixer idees noves que inicialment no havien estat planificades. Per falta de temps, no han estat implementades a la versió final.

Entre aquestes idees noves trobem la idea de fer possible un sistema de notificacions que avisi als usuaris cada cop que s'ha comentat una notícia seva, cada cop que es publiqui una notícia d'un usuari determinat, quan es voti una notícia, etc. Fer aquest sistema de notificacions és una tasca complexa i que necessita d'un servidor potent que pugui processar aquestes peticions.

A més, aquesta nova funcionalitat podria anar acompanyada d'un sistema que permetés seguir a diferents usuaris per estar al tant de les seves novetats. Aquest sistema de seguidors es pot implementar fàcilment, però no té cap sentit fer-ho si no es pot implementar abans el sistema de notificacions, ja que no serviria de res i no tindria efecte.

Aquestes són algunes idees noves que han aparegut durant el desenvolupament, però també n'hi ha d'altres que havien estat pensades al principi, tot i que no s'havien dissenyat per ser acabades durant el projecte, sinó que eren vistes a llarg termini. Aquestes idees inclouen la possibilitat de poder penjar una galeria de fotografies i vídeos sense necessitat d'estar associades a cap notícia. Es per això que els atributs de la Taula Media porten informació que a l'actualitat no s'utilitza.

També havia estat pensada la funcionalitat de que els usuaris poguessin tenir una puntuació i un rànking de manera interna. Això permetria que les notícies no sortissin a la pàgina inicial de manera automàtica, sinó que només sortirien les que tinguessin uns usuaris més fiables. I aquesta fiabilitat seria mesurada amb la seva puntuació i rànking intern. El rànking s'aniria definint a l'escriure diferents notícies i que aquestes rebessin bones puntuacions dels altres.

Com moltes altres funcions, aquesta també va ser pensada per a ser implementada a llarg termini, i es va preparar la base de dades per a la seva futura possible implementació.

Una altre idea és la de fer el sistema de registrar usuaris més robust, fent que sigui necessària una verificació per email de cada registre d'usuari. Aquesta idea comporta haver de fer una funció que envii els emails als nous usuaris i fer que la Taula userProfile compti amb un camp que seria si ha estat verificat i un altre que seria un codi per a poder fer aquesta verificació. La funció d'enviar els emails ja està feta a l'aplicació actualment, tot i que no s'utilitza en l'actualitat. El fet que aquesta funcionalitat no estigui implementada es degut a que el mapa era prioritari, i per tant tots els esforços van ser dedicats a fer que el mapa tingués un disseny acord a la web i que mostrés les diferents notícies. Llavors aquesta funcionalitat va ser aparcada per a ser implementada a mig termini.

Altres possibles millores a l'aplicació inclouen fer més robusta la seguretat de l'aplicació. A l'actualitat la base de dades està xifrada, i les contrasenyes no poden ser vistes de manera fàcil ja que estan encriptades, però sempre es pot millorar la seguretat.

Finalment també es podria fer una nova funcionalitat que faria canviar notablement el funcionament de la xarxa social. Es tractaria de fer un recomanador de notícies depenent dels gustos dels usuaris. Aquesta funcionalitat implicaria dissenyar i implementar algorismes d'intel·ligència artificial i per falta de temps ha estat impossible fer-la.

9 - Referències bibliogràfiques

^[1] **Menéame**

Menéame Comunicacions, SL (2005)

<http://www.meneame.net>

^[2] **La Tafari**

Tirabol Produccions, SL (2007)

<http://latafana.cat>

^[3] **Synkratos**

Synkratos (2013)

<http://www.synkratos.com>

^[4] **Estadístiques d'ús de navegadors**

StatCounter (2014)

<http://gs.statcounter.com>

^[5] **Pàgina web oficial de PHP**

The PHP Group (2001)

<http://www.php.net>

^[6] **What is HTML?**

World Wide Web Consortium, W3C (1993)

<http://www.w3.org/html/>

^[7] **W3Schools Online Web Tutorials**

Refsnes Data (1999)

<http://www.w3schools.com>

^[8] **Cascading Style Sheets home page**

World Wide Web Consortium, W3C (1996)

<http://www.w3.org/Style/CSS/>

^[9] **Pàgina web oficial de JavaScript**

Mozilla Developer Network (2005)

<http://developer.mozilla.org/es/docs/JavaScript>

^[10] **jQuery: write less, do more**

The jQuery Foundation (2006)

<http://jquery.com>

^[11] **Pàgina web oficial de Python**

Python Software Foundation (1991)

<https://www.python.org>

^[12] **Pàgina web oficial de MySQL**

Oracle Corporation (1995)

<http://www.mysql.com>

[13] Pàgina web oficial del framework Django

Django Software Foundation (2005)

<https://www.djangoproject.com>**[14] Tutorial de Django**

Cibernatural (2012)

<http://www.cibernatural.com/tutorial-de-django-i>**[15] Estadístiques sobre compatibilitat de navegadors amb HTML5 i CSS3**

Deep Blue Sky (2014)

<http://fmbjp.com/litmus>**[16] Pàgina web oficial de OpenShift**

Red Hat (2011)

<https://www.openshift.com>**[17] Pàgina web oficial de GitHub**

GitHub (2006)

<https://github.com>**[18] FontAwesome**

Dave Gandy

<http://fontawesome.github.io/Font-Awesome/>**[19] Like Button**

Facebook

<https://developers.facebook.com/docs/plugins/like-button>**[20] Twitter Buttons**

Twitter

<https://about.twitter.com/resources/buttons>**[21] Google+ Share Button**

Google (2013)

<https://developers.google.com/+/web/share/?hl=es>**[22] Tumblr Button**

Tumblr, Inc

<http://www.tumblr.com/buttons>**[23] Pinterest Button**

Pinterest

http://business.pinterest.com/en/widget-builder#do_pin_it_button**[24] Day Night Overlay**

Martin Matysiak (2011)

<https://code.google.com/p/google-maps-utility-library-v3/wiki/Libraries>

[25] jQuery File Upload Plugin

Sebastian Tschan (2010)

<https://github.com/blueimp/jQuery-File-Upload>**[26] Downward compatible, touchable dial**

Anthony Terrien (2012)

<http://anthonyterrien.com/knob/>**[27] Mini AJAX File Upload Form**

Martin Angelov (2013)

<http://tutorialzine.com/2013/05/mini-ajax-file-upload-form/>**[28] API de Google Maps**

Google (2013)

<https://developers.google.com/maps/>

10 - Annex

10.1 - Manual d'instal·lació

Com que Django^[13] és un framework que treballa sobre Python^[11] el primer que cal instal·lar és el propi Python. El software que instal·la Python de manera automàtica es pot descarregar des d'aquest enllaç: <https://www.python.org/download/>.

Una vegada ja tenim Python instal·lat cal instal·lar el framework Django, que pot ser descarregat al següent enllaç: <https://www.djangoproject.com/download/>.

La comanda per instal·lar Django per consola és, per a Windows:

```
python setup.py install
```

I per a Unix:

```
sudo python setup.py install
```

Django s'instal·larà automàticament al directori *site-packages* d'on estigui instal·lat Python. Per verificar que s'ha instal·lat de forma correcta podem fer, a través de l'interpret de Python, la següent comanda, sempre que estiguem en un directori diferent al de Django:

```
>>> import django
>>> django.VERSION
```

Si no dona cap error, i ens informa de la versió instal·lada de Django vol dir que està instal·lat de forma correcta. El següent pas es tenir una base de dades preparada per a poder ser utilitzada a l'aplicació. Podem fer servir diferents tipus de bases de dades.

Django suporta les següents bases de dades: PostgreSQL (<http://www.postgresql.org/>), SQLite 3 (<http://www.sqlite.org/>), MySQL (<http://www.mysql.com/>) i Oracle (<http://www.oracle.com/>). Per a l'instal·lació local de l'aplicació hem fet servir SQLite 3, que ja ve amb el projecte.

El següent pas és instal·lar la llibreria Pillow per poder veure i penjar imatges i vídeos: <http://pillow.readthedocs.org/en/latest/installation.html>.

Ara ja tenim la configuració inicial feta, el que cal és iniciar el servidor de Django. Ens situem al directori `wsgi/sharenews/` del projecte i executem la següent comanda.

```
python manage.py runserver
```

Això iniciarà el servidor i podrem fer accedir a l'aplicació.

Ara fa falta omplir la base de dades amb les primeres taules, per aconseguir això hem d'executar aquesta comanda:

```
python manage.py syncdb
```

Després d'executar-la ens demanà crear un super-user per poder accedir al menú d'administració. Ara sí, ara ja podem interactuar amb l'aplicació i fer totes les accions que ens ofereix.

10.2 - Manual d'usuari

Actualment es pot accedir a una versió online de l'aplicació sense haver-la d'instal·lar, es pot trobar en el següent enllaç: <http://tfg-sharenews.rhcloud.com>.

El manual d'usuari de l'aplicació està estructurat en com fer cadascuna de les diferents funcionalitats que permet fer l'aplicació.

Com registro un nou usuari?

La millor manera de registrar un nou usuari és a través de l'enllaç que podem trobar a la part superior de la pàgina web ("REGISTRA'T"). Aquest enllaç obrirà una nova pàgina on haurem d'omplir les dades que se'ns demanaran amb el format que ens indiquin.



REGISTRAT I PARTICIPA * Camps obligatoris

Nom

Cognom

Nom d'usuari * Entre 4 i 20 caràcters, només lletres bàsiques, números i _

E-mail *

Contrasenya * Entre 8 i 30 caràcters, només lletres bàsiques, números i @ . + - _

Repeteix la contrasenya *

He llegit i accepto les [condicions d'ús](#). *

ENVIAR

Fig. 31: Formulari per registrar un nou usuari

Com em connecto a la web?

Per connectar-nos a la web estant desconnectats ens hem de dirigir a la pàgina inicial de la web. Allà trobarem un formulari a la part dreta de la pàgina per poder-nos connectar. Només haurem d'entrar el nostre username i la nostra contrasenya per entrar.



CONNECTA'T Nou usuari?

Usuari

Contrasenya

ENTRAR

Fig. 32: Formulari per connectar-nos a la pàgina web

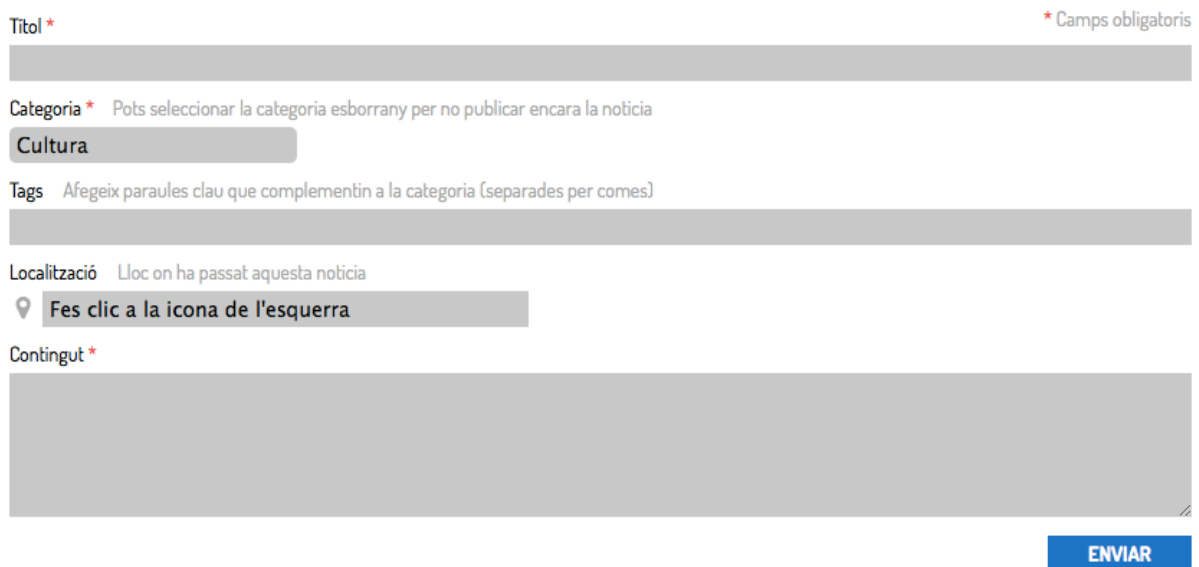
Com em desconnecto de la web?

A la barra superior de la pàgina web trobarem, si estem connectats, un enllaç que porta a la funció de desconnectar-nos (“DESCONNECTA’T”). Al fer clic sobre aquest enllaç ens desconnectarem automàticament de la pagina web.

Com redacto una nova noticia?

A la barra superior de la pàgina web trobarem, si estem connectats, un enllaç que porta a la pàgina de redactar una nova noticia (“REDACTAR”).

Al fer clic sobre aquest enllaç ens apareixerà una pàgina des de la qual podrem redactar una nova noticia introduint totes les dades que se’ns demanen. Obligatòriament hem d’escriure un títol, un contingut i assignar a la noticia una categoria. Les altres dades son opcionals.



Formulari per redactar una nova noticia. Els camps obligatoris estan marcats amb un asterisc (*).

Títol * * Camps obligatoris

Categoria * Pots seleccionar la categoria esborrany per no publicar encara la noticia

Cultura

Tags Afegeix paraules clau que complementin a la categoria (separades per comes)

Localització Lloc on ha passat aquesta noticia

Fes clic a la icona de l'esquerra

Contingut *

ENVIAR

Fig. 33: Formulari per redactar una nova noticia

Mentre estem escrivint la noticia podrem previsualitzar a la part inferior com quedarà una vegada estigui publicada a la web.

Com llegeixo una noticia?

Llegir una noticia és tan simple com fer clic al títol d'aquella que volem llegir. Això es pot fer en totes les pàgines on apareix alguna noticia.

Com puc veure fotos o vídeos?

A la vegada que estem llegint una noticia, podrem veure, si aquesta té alguna imatge o vídeo associat, un requadre on hi apareixen. Al fer clic en algun d'ells se'ns obrirà un panell on poder veure la imatge en gran o el vídeo.



Fig. 34: Panell per veure en gran les imatges i vídeos

Com puc puntuar notícies?

A la vegada que estem llegint una notícia, podem veure en la barra de la dreta un panell on poder puntuar la notícia.

Després de puntuar la notícia serem redirigits de nou a la notícia on podem fer més accions.

Puntuació actual: **3,33 / 5,00**



Fig. 35: Panell per puntuar una notícia

Com puc comentar notícies?

A la vegada que estem llegint una notícia, podem veure en la barra de la dreta, a la part de sota de la notícia, un panell on poder escriure un nou comentari.

Després de puntuar la notícia serem redirigits de nou a la notícia on podem fer més accions, i veurem com el nostre comentari està publicat.

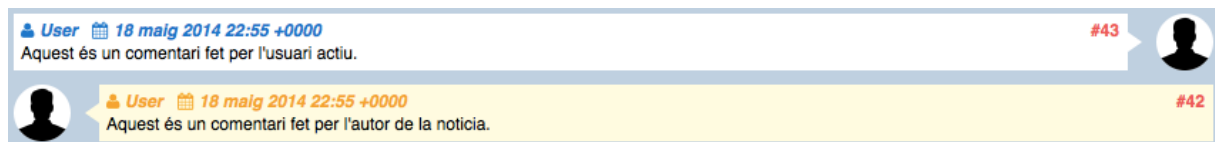


Fig. 36: Comentaris d'una notícia

Si el comentari apareix en groc vol dir que l'ha escrit l'autor de la notícia, i el comentari apareix alineat a la dreta en compte de a l'esquerra vol dir que és un comentari que ha fet l'usuari connectat.

Com puc compartir una notícia a una xarxa social?

A la vegada que estem llegint una notícia, podem veure un conjunt de botons on al fer clic obriran petits panells adaptats a cada xarxa social on podem ficar les dades per tal de poder compartir a les xarxes social.

L'aplicació permet actualment compartir a Facebook, Twitter, Google+, Tumblr i Pinterest.



Fig. 37: Botons per compartir a la xarxes social

Com puc penjar fotos i vídeos?

Mentre ens trobem redactant una nova notícia veurem a que a la part dreta hi ha un panell on podem penjar els arxius. Hi ha dues maneres diferents de penjar les fotos i vídeos: la primera és arrossegant l'arxiu al panell, i la segona és fent clic al botó "BUSCAR" per tal de seleccionar els arxius que vulguem penjar.

Una vegada estiguin seleccionats o els haguem arrossegats els penjaran automàticament i quedaran associats a la notícia que s'està escrivint. Si despès la notícia es descarta, els arxius també, i no es podran utilitzar.



Fig. 38: Panell per penjar imatges i vídeos

Com puc veure els perfils dels altres usuaris?

Per accedir al perfil dels altres usuaris només cal fer clic a qualsevol lloc on hi apareix el nom de l'usuari en qüestió, ja sigui en les notícies o en els comentaris.

Com puc actualitzar el perfil personal?

Si estem al nostre propi perfil veurem a la part de la dreta un botó per tal de modificar el perfil. Al fer clic se'ns obrirà un nou panell on poder ficar les noves dades actualitzades del perfil.



Fig. 39: Botó per actualitzar les dades del perfil

Com puc actualitzar el perfil personal?

Si estem al nostre propi perfil veurem a la part de la dreta un botó per tal de modificar el perfil. Al fer clic se'ns obrirà un nou panell on apareix un requadre on poder arrossegar la nova foto o poder buscar-la al nostre sistema i després es penjarà automàticament.

Com puc veure un mapa de notícies?

Hi ah diferents maneres de veure el mapa de notícies. A totes les pàgines on es mostren llistes de notícies es pot veure un minimapa que marca tots els punts de les notícies que tenen localització. A la pàgina inicial es mostren totes, a la de les diferents categories només les d'aquella categoria en concret, a la pàgina de buscar per tags només aquelles que tinguin els tags buscats, etc.



Fig. 40: Minimapa de la pàgina inicial

Una altre manera de veure el mapa és a través de l'enllaç que apareix a la barra superior. Aquest enllaç ens portarà a un mapa a pantalla completa on poder veure totes les notícies amb localització del moment.

També surt un minimapa a la pàgina de llegir una noticia si aquesta té localització. Aquest minimapa només mostrarà la localització d'aquella noticia.

Finalment, al perfil de cada usuari es pot veure també un minimapa indicant les localitzacions des d'on ha escrit notícies, sempre que aquestes tinguin localització.

Com puc filtrar notícies per categoria?

Per veure només les notícies d'una categoria en concret haurem de fer clic sobre aquella categoria que vulguem veure a la barra de la part superior de la pàgina. Se'ns obrirà una pagina molt similar a la pàgina inicial però només mostrant les notícies de la categoria seleccionada.

Com puc filtrar notícies per tags?

Per veure només les notícies amb uns tags en concret haurem d'escriure'ls al buscador de la part superior de la pàgina web, separant per comes cada tag que vulguem buscar. Se'ns obrirà una pagina molt similar a la pàgina inicial però només mostrant les notícies que tinguin els tags buscats.