



Treball de Fi de Grau

GRAU D'ENGINYERIA INFORMÀTICA

**Facultat de Matemàtiques
Universitat de Barcelona**

**DISSENY D'UNA XARXA SOCIAL ANDROID
PER COMPARTIR NOTÍCIES**

Gerard Reig Grau

Director: Dra. Maite López Sánchez
Realitzat a: Departament de Matemàtica
Aplicada i Anàlisi. UB

Barcelona, 18 de juny de 2014

Resum

El projecte que es desenvoluparà i s'explicarà al llarg de la memòria es una xarxa social de notícies la qual l'usuari pot publicar els esdeveniments que succeïxen en el seu entorn i explicar-les al món des del seu punt de vista, sense que aquestes siguin manipulades per fonts externes.

Avui en dia els medis de comunicació són manipulats o no són imparcials, tots ells tenen alguna connexió amb el sector polític o estan subvencionats per empreses privades. Així doncs, es pot arribar a la conclusió que qualsevol esdeveniment que succeïx no es transmet d'una manera objectiva.

Veient la situació actual és inevitable pensar com és possible que amb tota la comunicació que avui en dia existeix encara no sigui possible que una persona no pugui informar-se d'una manera veritable del que està passant al món.

El projecte estarà dividit en dos parts. La primera serà tota la relacionada amb la web, l'usuari es connecta mitjançant l'ordinador. Aquesta part es realitza en un altre Treball de Fi de Grau.

La segona part, la que s'explicarà i podrà llegir el lector, consistirà en la part de connexió des del mòbil, i concretament Android.

Abstract

The Project that will be developed and explained through the memory is a news based social media where the user can publish any event that is happening or recently happened in his environment, to tell the world from his point of view and without being changed or manipulated by any external source.

In the present day all media is controlled and/or biased as every TV station or news paper has some sort of link to certain political parties or private corporations. Therefore, we can conclude that the news and reports we get from conventional media are no objective and many times omit relevant information.

Seeing this situation, we do not understand how is it possible that with the world broadness of communication technologies we experience today we cannot access to reliable information of what is going on around the world.

The project will be divided in two parts. The first one will hold everything related to the World Wide Web (Internet), where the user will use a computer to get connected into the program. This part is done in another Undergraduate Final Project (or TFG “Treball de Final de Grau”). The second part, the one that will be explained and readable for the audience; will consist in the sections of the program regarding portable devices with “Android” technology.

Index

1. Introducció i motivació	6
2. Objectius	7
3. Estudi de mercat	9
3.1 Android	9
3.2 Producte	10
3.2.1 Cicle de vida mercat	10
3.2.2 Competidors	11
3.2.3 Missió i visió	14
3.2.3.1 Missió	14
3.2.3.2 Visió	14
3.2.4 Anàlisi DAFO	14
3.2.4.1 Debilitats	14
3.2.4.2 Amenaces	14
3.2.4.3 Fortaleses	14
3.2.4.4 Oportunitats	15
3.2.5 Estratègies	15
3.2.5.1 Estratègia seguidor	15
3.2.5.2 Estratègia de diferenciació	16
3.2.6 Model negoci	17
4. Arquitectura	18
4.1 Servidor	18
4.1.1 OpenShift	18
4.1.2 GitHub	19
4.1.2.1 Avantatges al penjar el codi	20
4.1.2.2 Característiques útils	21
4.1.2.3 Revisió de canvis en el codi	21
4.1.2.3 Preu	22
4.1.3 Framework Django	22
4.1.3.1 Estructura	23
4.1.3.2 Classes	23
4.1.3.3 Suport de la base de dades	23
4.1.3.4 Proces d'una request	24
4.1.4 Base de dades	24
4.1.4.1 Diagrama ER	26
4.1.4.2 Descripció Entitats de la taula	26
4.2 Android	29
4.2.1 Classes usades	29
4.2.1.1 Activity	29
4.2.1.2 Fragment	30
4.2.1.3 Adapter	31
4.2.1.4 Runnable vs Thread	31
4.2.2 Connexió amb el servidor	31
4.2.2.1 Com passar informació per la request	32
4.2.2.2 Json	33
4.2.2.3 MultipartEntityBuilder	34
4.2.3 Google Maps	34

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

<u>5 Disseny</u>	36
<u>5.1 Funcionalitats</u>	36
<u>5.2 Casos d'us</u>	36
<u>5.3 Casos d'us textuais</u>	37
<u>5.4 Patró MVC</u>	44
<u>6. Desenvolupament</u>	46
<u>6.1 Programació</u>	46
<u>6.1.1 Funcions Views</u>	46
<u>6.2.2 Classes Android</u>	47
<u>7. Resultats i test</u>	49
<u>7.1 Usuaris</u>	49
<u>8. Estudi econòmic</u>	51
<u>9. Conclusions i treball futur</u>	53
<u>9.1 Conclusions</u>	53
<u>9.2 Treball Futur</u>	53
<u>10. Referències bibliogràfiques</u>	54
<u>11. Annex</u>	57
<u>11.1 Manual d'usuari</u>	57
<u>11.2 Manual d'instal·lació</u>	63
<u>11.2.1 Eclipse instal·lació</u>	63
<u>11.2.2 Google Maps Instal·lació</u>	65

1. Introducció i motivació

Tal com he dit anteriorment tot va sorgir d'una assignatura que vam cursar a la carrera. Per entendre el perquè daixò vull explicar dos histories que van ser causants de la decisió de fer aquest projecte, potser no son importants per al lector però per mi aquestes histories son el desencadenant d'aquest treball. En la meua humil opinió els grans projectes i idees sempre s'han degut a fets concrets en la vida del creador els quals l'han fet reflexionar si podia fer alguna cosa al respecte.

La primera historia es remunta a ara farà uns tres anys. Un estiu vaig anar amb els amics a passar una setmana a Mallorca, allà vivia el pare d'un del grup el qual treballava a la televisió autonòmica d'allà, ens va acollir a casa seva i ens va donar de menjar, transport, rutes per veure les cales i tot lo relacionat perquè tinguéssim uns bons dies.

Una nit mentre estàvem sopant va sortir al tema que aviat el pare tornaria a Barcelona perquè s'havia quedat sense feina. Al sentir això el primer que se'm va passar pel cap va ser preguntar-li a que es devia això, que com es que ara després de tan de temps el feien fora si era una persona responsable i es duia be amb tothom de la feina, i la resposta que em va dir em va deixar atònic. Simplement havien guanyat les eleccions el PP i al haver un canvi de govern van fer fora a tota la plantilla per substituir a tot l'equip per un de nou decidit per el govern. El que potser per al lector li sembla normal (per mi no ho es) en aquell moment em va deixar de pedra. No entenia que tenia a veure el govern amb les notícies, les quals son un mitja de comunicació i difusió dels esdeveniments independentment de l'estat, la situació, etc.

La segona historia que em va motivar a fer aquest projecte es remunta a no mes de mig any. Suposo que el lector coneixerà els problemes que hi ha hagut a Ucraïna. No explicaré ara tot això nomes vull donar a entendre la situació. Amb una amiga que es d'allà sempre parlàvem del que estava passant, que opinava, etc. El problema es que el meu punt de vista es basava amb la informació que jo rebia de les notícies, diaris d'internet i altres fonts de difusió privades. Ella sempre em deia que el que veia no era res del que realment estava passant allà i que la informació que sens donava aquí era manipulada o directament falsa. Llavors amb que hem podia basar per treure la meua opinió si la informació que m'arribava era mes que qüestionable? Jo encara que coneixia una persona d'Ucraïna que m'explicava de primera ma el que vivien els seus familiar allà, però i la gent que vol saber que passa realment? Don pot treure un lloc fiable informació?

Tot això em va fer recapacitar si realment estàvem tan ben informats com el que sens diu o sens vol fer creure. Don podia treure la informació i veure que passa al mon?

2. Objectius

L'objectiu principal es la creació d'una xarxa social de notícies. Aquest projecte té la intenció de convertir-se en un lloc de difusió d'informació. L'usuari podrà llegir els esdeveniments que succeeixen en el món d'una manera clara, senzilla i sense tecnicismes, i a la vegada ser ell mateix el propi redactor dels fets i esdeveniments que passen en seu entorn.

Es vol transmetre la idea que hi ha un altre font d'informació que no pas els ja coneguts i manipulats, on una altra persona com ella ha escrit la notícia i li està explicant des de el seu punt de vista.

Creiem que els medis de comunicació no són actualment una font de fiabilitat per la població. Són ja molts casos en que aquests no han transmes tota la informació de la notícia o directament han fet veure que un esdeveniment mai havia passat.

L'aplicació estarà formada per una part web i una altra part per a mòbil. Així doncs el projecte estarà dividit en dos parts remarcades, les dos tindran que estar connectades entre si per compartir l'informació i cada una de elles haurà de tenir les funcionalitats d'una i altra. Hauré d'utilitzar un servidor per emmagatzemar tota la informació i crear la lògica necessària per poder utilitzar els dos recursos el paral·lel sense que hi hagi errors.

La part web és un projecte desenvolupat per el meu company el qual utilitza diferents llenguatges de programació web per a crear una interfície d'usuari simple e intuïtiva. Tota la informació d'aquest projecte es pot llegir en la memòria en qüestió, en aquesta memòria no s'explicarà la part web però potser en algun moment es menciona.

En el projecte en qüestió s'ha desenvolupat la part relacionada amb el mòbil i concretament android. Al llarg de la memòria s'explicarà les decisions que s'han pres, com s'han arribat a la conclusió d'aquestes i les funcionalitats dutes a terme.

Es va decidir fer la interfície pel mòbil perquè es considerava que era més natural publicar les notícies just en el lloc on succeeixen aquestes, que l'usuari no hagués d'anar a casa a escriure tot el que estava veient, sinó que just al moment en que passava fos capaç de transmetre al món, fer dos o tres fotografies, un vídeo i poder-ho publicar.

A més, avui en dia d'utilització del mòbil ha estat incrementada en comparació anys endarrere i cada dia que passa aquest augmenta encara més. Les millores de rendiment i les capacitats d'aquest han fet aparèixer un nou mercat on la vida social ja es desenvolupa des de els mòbils.

Per acabar esmentar que per la part acadèmica al llarg del projecte s'han utilitzat recursos de moltes assignatures que s'han cursat durant aquests quatre anys de la carrera i que gràcies a elles ens han servit de base per a poder començar cada una de les parts diferents les quals està format el projecte. Aquí una llista de les utilitzades: *Programació I i II, Algorísmica i Algorísmica Avançada,*

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

Estructura de Dades, Disseny de Software, Projecte Integrat de Software, Sistemes Operatius I i II, Xarxes, Taller de Nous Usos de la Informàtica, Intel·ligència Artificial, Gràfics i Visualització de Dades, Factors Humans i Computació, Bases de Dades, Software Distribuït i Enginyeria del Software.

Però de totes les assignatures potser la que més s'ha vist el símil es la de “Software Distribuït” i exactament la practica tres, la qual consistia en fer una casa d'apostes. El servlet en si es el framework Django el qual explicaré més endavant. Lògicament “Projecte Integrat de Software” també ha estat una bona base per a començar el projecte.

3. Estudi de mercat

Avanç de tot es farà un petit estudi de mercat relacionat amb Android i després amb pàgines webs relacionades amb el projecte. Aquest servirà per determinar la situació del mercat i les possibilitats del nostre projecte en relació als que es troben ja en funcionament. Qualsevol empresa quan decideix treure un producte en un mercat ha de fer un estudi com aquest.

A la part de l'anàlisi de la plataforma Android buscaré informació sobre la utilització d'aquesta pels usuaris i les diferents versions que es troben actualment en el mercat.

La part de l'estudi de mercat en relació al projecte serà més detallada i analitzada ja que el projecte en si s'ha de moure en aquell mercat, la plataforma pot canviar i segurament en pocs anys així sigui.

Al llarg de l'estudi s'utilitzarà terminologia que no prové de la Enginyeria Informàtica sinó de Administració D'Empresa, tot i així són conceptes molt bàsics que el lector podrà entendre, i si hi hagués alguna terminologia molt concreta s'explicaria que es d'una manera senzilla, clara i bàsica

Per acabar dir que es tractarà el projecte com un producte, ja que aquest en si no es diferencia en res d'un producte físic que una empresa vulgui vendre, és més, nosaltres l'estem venent però no treiem benefici d'ell directament. També la descripció dels usuaris pot ser com de clients o vendes.

3.1 Android

Tota la informació que a continuació es mostrarà i els resultats que analitzarem han estat obtinguts de NETMARKETSHARE (Market Share Statistics for Internet Technologies) i analitzats al 11 de Març.

Pel que fa a la utilització dels diferents sistemes operatius per a mòbil hi ha dos forces que es superposen a les altres, una és iOS i l'altra Android. La primera ocupa un major nombre en el mercat, un fet que es diferencia d'anys anteriors on Android era avanç la primera força. (Figura 1. Sistemes Operatius mòbil)

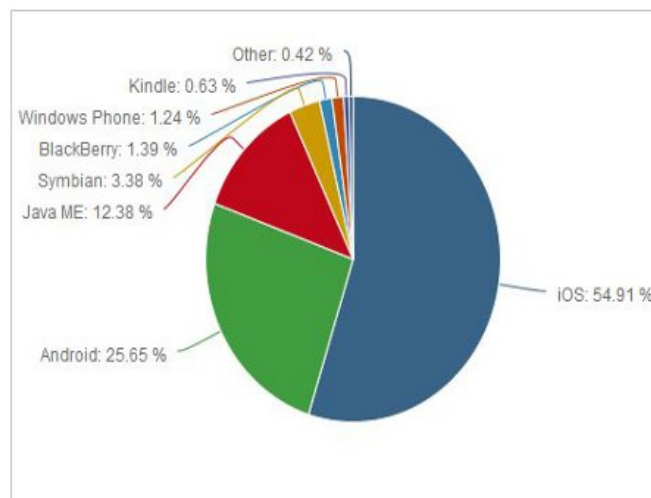


FIGURA 1. Sistemes operatius mòbil

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

No obstant això, la tendència no és molt positiva per iOS, que ha perdut gairebé un 10% de terreny, passant del 63,19% a l'abril 2012 a un 54,91% al febrer 2013. En canvi Android ha guanyat part del terreny perdut per iOS, passant d'un 19,27% a un 25,65% al febrer. Java EM es manté, mentre que Symbian mostra un creixement inesperat després d'un 2012 de baixes. Blackberry descendeix d'un 2.04 a un 1.39%.

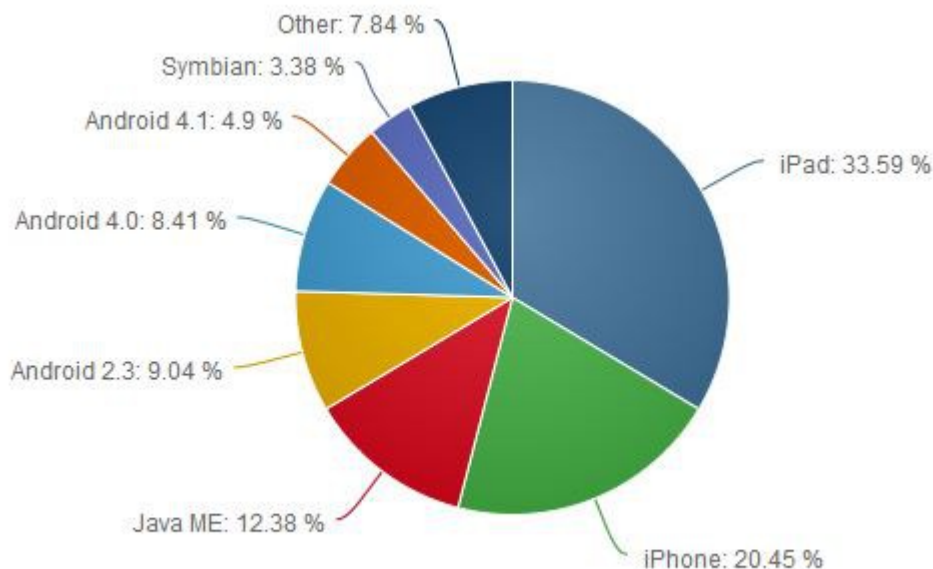


FIGURA 2. Versions Android

Per versions (Figura 2. Versions Android), les tendències mostren que la reculada de iOS és a causa de l'iPhone ja que l'iPad és un valor segur, mantenint-se en el 33,59%. El iPhone ha retrocedit des d'un 27,36% a l'abril 2012 a un 20,45% al febrer, indicant una clara tendència a la baixa. En Android, la versió 2.3 va deixant terrè a la versió 4.0 que passa d'un 0,94% a l'abril 2012 a un 8,41% al febrer 2013. Windows phone 8 continua a poc a poc menjant terrè, sobretot a Blackberry.

3.2 Producte

3.2.1 Cicle de vida mercat

Per a qualsevol persona que navegui una mica per internet sabrà que pàgines webs relacionades amb notícies n'està ple, hi ha varies opcions per al usuari, des de webs de grans multinacionals, webs de diaris, webs de cadenes de televisió, fins a opcions més senzilles com blogs de persones les quals expliquen els esdeveniments que succeïxen en el seu entorn però la majoria d'elles, per no dir casi totes, l'usuari no pot redactar la notícia, simplement la llegeix d'un portal on son agrupats uns quants periodistes, redactors, etc. Els quals son els encarregats de escriure-les.

El que està clar, és que és un mercat molt explotat i que es troba en la etapa de maduració. (Figura 3. Cicle de Vida)

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

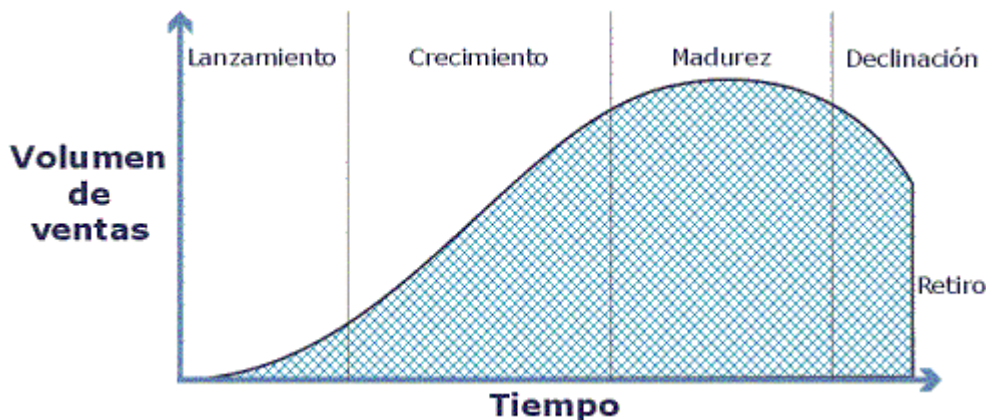


Figura 3 Cicle de vida

Quan el producte ha aconseguit la màxima participació possible al mercat, el volum de vendes comença a estabilitzar-se, la majoria de les vendes es dirigeixen a usuaris reiterats i els costos, probablement, han seguit reduint-se. Inicialment, els recursos generats pel producte són alts i l'empresa comença a recollir els seus fruits en deixar de necessitar una gran inversió per augmentar o mantenir la posició assolida. Les posicions de mercat obtingudes per competidors ben posicionats dificulten l'ingres de nous participants com es el cas del nostre producte.

Algunes de les característiques pròpies d'aquesta etapa son:

- òptim nivell de penetració de mercat
- poc o nul creixement de la demanda
- màxima rendibilitat i contribució possibles
- major desenvolupament de la segmentació i nínxols
- màxima acció de la competència per desplaçar posicions aconseguides

3.2.2 Competidors

A continuació es farà un analitzi dels competidors que es poden trobar a internet els quals podrien fer disminuir la cota de mercat. Per superar als rivals es important saber quins son els seus punts forts i dèbils , i inclús aconseguir idees per al producte veient quines son les cadències que tenen les altres webs. s'elegiran uns quants exemples per mostrar el que un usuari qualsevol pot trobar a internet en busca de notícies:

<http://news.google.es/>

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES



Figura 4. News Google

La plataforma Google (Figura 4. News Google) te un portal on agrupa les notícies de diferents diaris fent que l'usuari pugui accedir d'una manera ràpida i senzilla. L'usuari pot fer una selecció de notícies que vol veure en funció unes característiques com la categoria, el nombre de visites o la data de publicació. Una fortalesa que te es la personalització de cerca que te desenvolupada, dona a l'usuari un gran ventall de possibilitats.

Tot i així no enfoca la idea del projecte fent que no sigui un competidor almenys a curt termini.

<http://www.eleconomista.es/>



Figura 5 elEconomista

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

Un altre exemple de portal de notícies es aquest. L'usuari pot trobar moltes notícies dividides en categories, utilitzar els tags, i les funcionalitats habituals que acostumen a tenir aquestes pàgines però no pot l'usuari crear les notícies. (Figura 5. elEconomista.es)

Posseeix una aplicació mòbil de la pàgina web, un factor a tenir en compte.

<http://www.synkratos.com/>



Figura 6. Synkratos

Aquest es clarament el principal competidor que hi ha. Es un projecte en desenvolupament format per 6 membres i que ja tenen un pressupost de 4.000 euros que han aconseguit gràcies a donacions d'usuaris els quals volien que es formés el projecte. Tenen la idea clara i sembla ser que donaran les mateixes funcionalitats que l'aplicació vol oferir. No es pot saber gaire més informació ja que encara està en construcció. (Figura 6. Synkratos)

Un punt a favor que té el projecte és la possibilitat de sortir al mercat avanç que el competidor tot i així s'ha d'anar en compte perquè el fet de treure el producte avanç pot implicar que no es compleixin els mínims de qualitat necessaris comportant la lapidació el nostre projecte avanç de poder-se donar a conèixer.

En qualsevol cas aquest seria el líder en qüestió el qual s'hauria de superar.

3.2.3 Missió i visió

3.2.3.1 Missió

La definició de missió d'una empresa respon a la pregunta “Què som?”. 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.

3.2.3.2 Visió

La definició de visió d'una empresa respon a la pregunta “Què volem ser?”. Volem convertir-nos en un portal que ofereixi la oportunitat a la població de poder saber la realitat què 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.

3.2.4 Anàlisis DAFO

Les sigles DAFO provenen de les paraules “Debilitat, Amenaça, Fortalesa, Oportunitat”. L'anàlisi DAFO es una metodologia d'estudi de la situació d'una empresa o un projecte analitzant les seves característiques internes (Debilitat, Fortalesa) i la seva situació externa (Amenaça, Oportunitat). Aquesta eina serveix per conèixer la situació real de la empresa o projecte en qüestió i idear una estratègia pel futur.

3.2.4.1 Debilitats

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

3.2.4.2 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.

3.2.4.3 Fortaleses

- Cobrim una necessitat del mercat existent.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

- 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).

3.2.4.4 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.
- Cobrim una carència del mercat.

Potser els punts més importants com a Fortaleses que té el projecte són el fet que l'usuari pot escriure la seva pròpia notícia i que el producte disposi de un mapa on l'usuari pot interactuar i veure la localització de les notícies en el món.

El gran avantatge que la diferencia de la majoria dels altres productes i el que s'espera que els clients acceptin és el fet de que poden escriure les notícies ells. Les altres webs no donen aquesta opció.

Un altre tret característic és la utilització del mapa Google. Aquest permet a l'usuari saber la localització de la notícia en el món real, o buscar notícies basant-se en el mapa. Aquest factor no el tenia ni principal competidor Synkratos. Així doncs, s'haurà d'explotar al màxim aquest atribut ja que és el que pot fer assumir la primera posició en la cota de mercat.

3.2.5 Estratègies

Analitzant el DAFO, els competidors i la situació del mercat s'opta per la utilització de dos estratègies, la de seguidor i la diferenciació. Creiem que amb aquestes podrem començar a tenir presència en el mercat actual i impulsar el nostre producte.

3.2.5.1 Estratègia seguidor

Aquesta estratègia és habitualment adoptada per al competidor que té una quota de mercat més reduïda que el líder. Consisteix a alinear les seves decisions pel que fa a les del capdavanter. No ataca, coexisteix amb ell per repartir-se el mercat. Tracta de desenvolupar la demanda genèrica concentrant-se en segments del mercat en els quals posseeix un major avantatge competitiu, amb una estratègia pròpia.

Com que el projecte està encara en formació i creixement es poden adoptar moltes mesures que els competidors ja utilitzen degut a la seva experiència en el mercat i el temps que porten. No cal

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

invertir temps ni recursos en solucionar problemes que es poden trobar si ja hi ha projectes d'altres empreses que els solucionen utilitzant diferents eines. El que farà es mirar les diferents opcions que hi ha i s'elegirà la millor o inclús un mix d'elles.

També s'han de tenir clar les limitacions de cada empresa. L'empresa esta formada per dos empleats, els quals es troben per primer cop en un mercat nou per ells i utilitzen varies eines diferents i noves que aprenen a utilitzar-les a la vegada que en fan us d'elles. S'ha de saber administrar el temps, decidir en quins aspectes s'ha de concentrar, i que s'ha de fer per diferenciar-nos dels competidors amb els pocs recursos que tenim. Això comporta a la segona estratègia que s'adoptarà.

3.2.5.2 Estratègia de diferenciació

Es tracta de donar-li al producte un atribut que sigui percebut per la indústria com a únic. Per fer-ho la empresa selecciona un atribut del producte el qual el mercat considera que es important i el desenvolupa per a satisfer la demanda dels clients (Figura 7. Estratègies).



Figura 7. Estratègies

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

L'objectiu de l'aplicació es que sigui utilitzada per usuaris de totes les edats. No importa que sigui un estudiant, un adult o una persona de la tercera edat. Es una xarxa social on qualsevol persona pot donar la seva visió de l'actualitat i comparar-la amb la dels altres, així doncs es decideix que l'objectiu es cobrir tot el sector.

Pel que fa l'àmbit de costos el considerem nul ja que l'únic factor que ens pot costar capital seria el servidor i la conta de Android per poder pujar les apk's.

Es te clar doncs que s'ha d'adoptar una estratègia de diferenciació, s'ha d'aconseguir alguna cosa que ens diferenciï dels altres competidors que ja porten mes temps i estan establerts còmodament en el mercat. S'ha de tenir en compte com s'ha mencionat avanç les limitacions que te l'empresa, així doncs es posaran d'objectius aquest punts que diferencien el producte i s'invertiran els esforços i temps allà.

Tal com s'ha mencionat abans el punt fort del producte que ens diferencia dels competidors es el fet que l'usuari es qui escriu la noticia, les altres webs no ho permeten. Un altre punt es el mapa que ens diferencia de tots els altres productes del mercat ara mateix.

3.2.6 Model negoci

Un cop decidida la estratègia que es seguira, s'analitza la situació econòmica que es necessita per poder fer funcionar el projecte. Gràcies a que s'han utilitzat servidors i llicències gratuïtes no es fa falta un pressupost inicial per a començar el projecte.

Com a font d'ingressos exteriors es considera que no es vendrà el producte en si sinó que els beneficis que s'aconsegueixin seran d'anuncis d'internet o contribucions voluntàries dels usuaris.

Tal com s'ha mencionat anteriorment l'aplicació no requerira pressupost per dur-se a terme i mantenir-se amb lo qual es creu que amb aquest pocs ingressos dels anuncis i contribucions que es puguin aconseguir ja seran suficients per a mantenir en un futur (si escau) un servidor de pagament que pugui garantir mes estabilitat i mes transit a la web.

4. Arquitectura

Un cop realitzat tot l'estudi de mercat i es te clar com es vol ambientar el projecte se disposa a estructurar-lo per diferenciar be que es cada part i la seva funcionalitat. També s'explicarà quins recursos utilitzen, el pros i contra.

Consistira en tres parts bàsicament:

- Web del projecte:** Aquesta part l'encarregat de fer-ho ha estat el meu company i amic David Ortega Lázaro el qual el seu TFG consisteix en fer la part de la web i la seva connexió respectiva amb la base de dades. No s'explicarà en aquesta memòria com funciona la seva part, tot i així potser en alguns moments al llarg de la explicació es mencioni per relacionar o comparar amb la meva part. Els llenguatges que ha utilitzat son HTML,CSS,JavaScript i Python per comunicar-se amb el servidor.

- Aplicació Android:** Aquesta part es la qual he estat l'encarregat de crear. Bàsicament ofereix les funcionalitats del projecte des de la plataforma Android. Es connecta al mateix servidor que el de la web.

- Servidor:** Contindrà la base de dades la qual esta escrita en mypsql. També disposara del framework Django el qual administrara les peticions d'entrada. En aquesta part depenen quina request li arriba farà una funcionalitat o altre. Es l'encarregat de fer possible la interacció entre la Web i Android

4.1 Servidor

Avanç d'explicar-se la part de Android es parlara del servidor i tot el que hi ha darrera d'aquest.

El primer pas que es va haver de decidir va ser la plataforma que administraria el servidor, el domini i la base de dades. Un requeriment que es demanava es que la interacció amb ell pogués ser amb python ja que es un llenguatge de programació còmode i basant-nos en la experiència es sabia que no hi hauria problemes per relacionar l' informació que arribes des de la web o android, per suposat també havia de ser gratuït, almenys per al inici del desenvolupament no es necessitava més.

Recordo que al primer any de carrera a l'assignatura Algorísmica 1 vaig preguntar al professor perquè servia el llenguatge Python si ja s'utilitzava Java i la resposta va ser que Python s'utilitzava com a enllaçador de diferents aplicacions, en aquell moment no sabia ni de que m'estava parlant però al buscar informació d'aplicacions de hosting em va venir al cap.

4.1.1 OpenShift

Buscant informació per internet es va trobar un hosting que oferia un framework que funcionava amb Python i pel que semblava proporcionava moltes eines útils que encara que en aquell moment no es necessites res de l'altre mon per dur a terme el projecte en un futur quan es volgués treure el projecte al mercat ens podrien servir.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

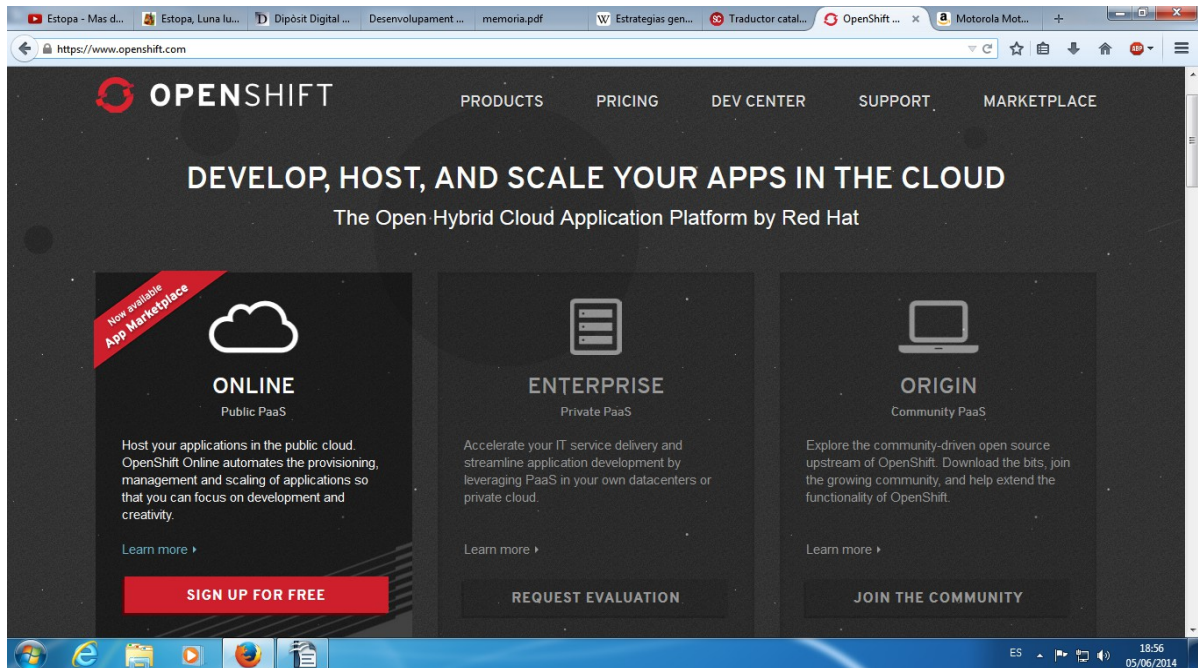


Figura 8. OpenShift

Es una plataforma d'aplicacions de cloud que automatitza l'allotjament, configuració, implementació i administració de piles d'aplicacions en un entorn de cloud flexible. Aporta accés d'autoservei als desenvolupadors d'aplicacions perquè puguin implementar fàcilment aplicacions sota demanda. (Figura 8. OpenShift)

Avantatges:

- Redueix el temps necessari per construir i penjar les aplicacions, fent així que l'usuari en concentri en el codi i les innovacions en comptes de les infraestructures i l'administració.
- Dona moltes opcions de llenguatge de programació, frameworks i runtimes incloent Java EE6 amb JBoss EAP.

Des de casa es pot accedir al servidor utilitzant el terminal Git Shell. Allà un cop feta la configuració que s'explicarà al final de la memòria sens dona els permisos per entrar a tots els continguts del servidor i poder veure els arxius. També es poden modificar els arxius des de aquí però una manera més còmode de fer-ho es mitjançant el GitHub el qual OpenShift permet utilitzar. Hi ha un petit tutorial a la web on indica els passos que l'usuari ha de seguir pe poder-ho utilitzar.

4.1.2 GitHub

GitHub es un servei de allotjament de repositoris de software utilitzant el sistema de control de versions Git. (Figura 9. GitHub web)

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

Es un servei de projectes de software lliure ja que els projectes que es pengen allà poden ser vistos per a tothom. Algunes grans empreses com Facebook allotgen allà els seus projectes públics com SDK, llibreries, exemples, etc. Comentar que totes les llibreries que he utilitzat per android estaven penjades al GitHub.

En el projecte s'ha utilitzat ja que la part de Web i Android utilitzen el mateix framework i base de dades, així cadascú des de on treballes podia anar fent els canvis necessaris per al correcte funcionament de la seva part.

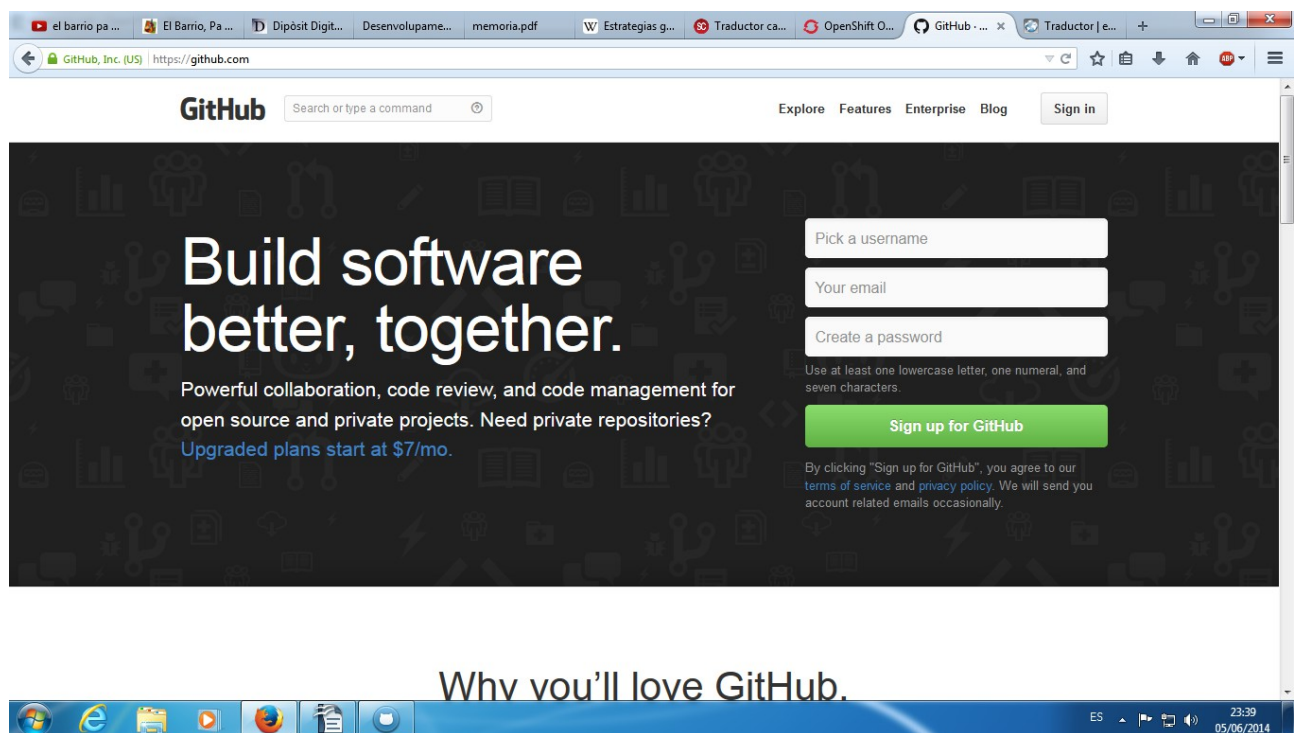


Figura 9. GitHub web

4.1.2.1 Avantatges al penjar el codi

Imaginem un projecte que esta format per varies persones les quals han de treballar sobre el mateix codi, seria impossible fer-ho a la vegada ja que si un realitza uns canvis en una pagina de codi, un cop ha finalitzat la feina ha de penjar el codi o passar-lo al seu company perquè modifiqui a partir de la nova versió.

GitHub permet a cada usuari treballar des de el seu codi i un cop finalitzat els canvis penjar-ho al repositori utilitzant el “commit”. El company de feina només haurà de “sincronitzar” el repositori des de on estigui treballant i les modificacions fetes per al primer seran afegides en el seu codi. (Figura 10. GitHub Historial)

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

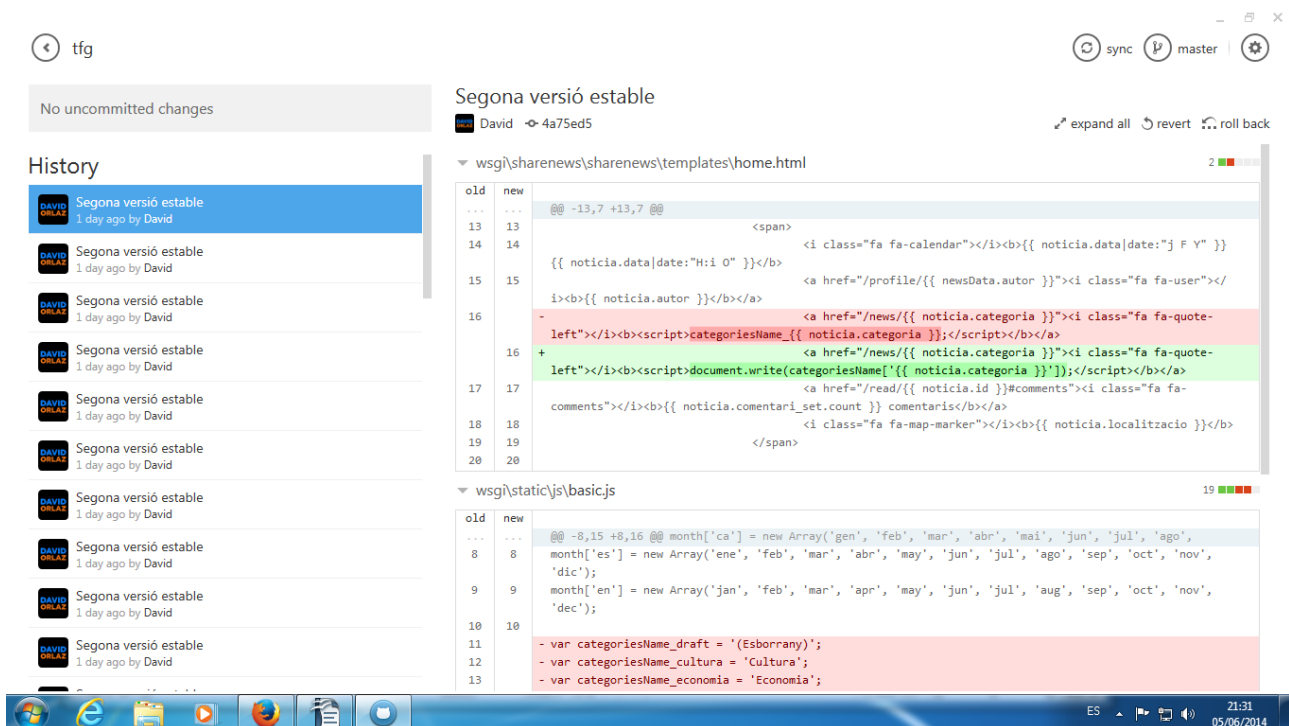


Figura 10. GitHub Historial

4.1.2.2 Característiques útils

També ofereix varies eines per millorar el treball en equip.

- Una wiki que funciona amb gollum, el qual funciona amb Git per al manteniment de las diferents versions de las pagines.
- Un sistema de seguiment de problemes que permet obrir un ticket detallant un problema que tingui el software o un suggeriment que es vulgui fer. Si el repositori esta en public això ho pot fer qualsevol usuari.
- Una eina de revisió de codi, on es poden afegir anotacions en qualsevol punt del fitxer i debatre sobre determinats canvis realitzats en un commit específic.
- Un visor de branques on es poden comparar els progressos realitzats en les diferents branques del nostre repositori.

4.1.2.3 Revisió de canvis en el codi

A més de treballar amb el teu equip en el software pots contribuir a millorar els softwares dels altres, aquestes funcions son el *fork* i *pull*. Funcionen de la següent manera: Imagina que mirant una llibreria lliure que algú ha fet i esta penjada al GitHub veus una manera de millorar el software corregint un error o optimitzant-lo. Pots clonar el repositori (fork) per a que es copii a la teva conta, fer els canvis que necessitis en la copia i finalitzar emetin una sol·licitud de *pull* al propietari del

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

repositori original. Aquest pot analitzar els canvis que s'han realitzat i si considera correcte la teva contribució fer un *merge* amb el seu repositori original.

4.1.2.3 Preu

Es totalment gratuït allotjar el codi open source, sent aquest repositoris de visibilitat pública obligatòriament. No obstant això, el seu pla de preus per als repositoris privats i amb col·laboradors externs és més que raonable, abastant des de 5 repositoris privats i fins a un col·laborador per 7 dòlars al mes, fins a 125 repositoris privats i equips de persones il·limitats per 200 dòlars al mes.

4.1.3 Framework Django

Django és un framework de desenvolupament web de codi obert, escrit en Python, que respecta el paradigma conegut com Model Template View. Va ser desenvolupat en origen per gestionar diverses pàgines orientades a notícies de la World Company de Lawrence, Kansas, i va ser publicada sota una llicència BSD al juliol de 2005. El framework va ser nomenat en al·lusió al guitarrista de jazz gitano Django Reinhardt.

Al juny del 2008 va ser anunciat que la recentment formada Django Programari Foundation es faria càrrec de Django en el futur.

La meta fonamental de Django és facilitar la creació de llocs web complexos. Django posa èmfasi en re-utilitat, la connectivitat i extensibilitat de components, el desenvolupament ràpid i el principi “No et repeteixis” (DRY, de l'anglès Don't Repeat Yourself). Python és usat en totes les parts del framework, fins i tot en configuracions, arxius, i en els models de dades

Algunes característiques que te:

- Un mapejador objecte-relacional.
- Aplicacions que es poden instal·lar-se a qualsevol pàgina gestionada amb Django.
- Una API de base de dades robusta.
- Un sistema incorporat de "vistes genèriques" que estalvia haver d'escriure la lògica de certes tasques comunes.
- Un sistema extensible de plantilles basat en etiquetes, amb herència de plantilles.
- Un dispatcher de URLs basat en expressions regulars.
- Un sistema "middleware" per desenvolupar característiques addicionals; per exemple, la distribució principal de Django inclou components middleware que proporcionen escorcoll, compressió de la sortida, normalització de URLs, protecció CSRF i suport de sessions.
- Suport d'internacionalització, incloent traduccions incorporades de la interfície d'administració.
- Documentació incorporada accessible a través de l'aplicació administrativa (incloent documentació generada automàticament dels models i les biblioteques de plantilles afegides per les aplicacions).

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

Per la part de Android el protocol habitual es que aquest envii una request a Django, aquest al rebre la request executa la funció de la view que correspongui i la llavors retornar una response a Android. Acostuma a retornar una informació codificada en “json” o en “File”, aquest últim seria una imatge o vídeo i el json s'explicarà més endavant a l'apartat Android el que es.

4.1.3.1 Estructura

Encara que Django està fortament inspirat en la filosofia de desenvolupament Modelo Vista Controlador, no segueix cap paradigma particular. Com a resultat, per exemple, la qual cosa s'anomenaria "controlador" en un "veritable" framework MVC es diu en Django "vista", i el que es diria "vista" es diu "plantilla". En el cas de voler posar-hi un nom, llavors diríem que es un framework “MTV”: *model, template, vista*.

4.1.3.2 Classes

URL dispatcher

Depenent la url que es crida del servidor, el dispatcher de Django s'encarrega de redirigir aquesta petició a la funció corresponent que es trobi a la view. Aquest arxiu es proporciona per Django i el desenvolupador només s'ha d'encarregar de fer les relacions que necessiti.

Views

La views es un arxiu on bàsicament hi ha totes les funcions que el desenvolupador hagi creat. Allà es on es duu a terme tota la lògica de la petició, com per exemple les consultes a la base de dades, operacions lògiques, eliminació, creació, etc.

Templates

La última part de la petició es el template. Tot i així no s'explicarà en aquesta memòria ja que en la part d'android aquests no s'utilitzen ja que no es mostren resultats en forma visual, només es retorna informació al mòbil.

Models

A més de tota la part relacionada amb les request hi ha un altre arxiu a tenir en compte, aquest es el models. Allà està definida tota la estructura de la base de dades com les taules, els atributs d'aquests i el format de valor que s'ha d'introduir a cada un dels atributs. També es poden posar valors per defecte. En algunes funcionalitats s'ha utilitzat aquest recurs, com per exemple en la creació de notícies, per exemple si no introduïa algun camp en concret.

4.1.3.3 Suport de la base de dades

Respecte a la base de dades, pot suportar PostgreSQL SQLite 3 i MySQL. En el cas del projecte s'ha utilitzat aquest últim.

Una vegada creats els data models, Django proporciona una abstracció de la base de dades a través

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

de la seva API que permet crear, recuperar, actualitzar i esborrar objectes. També és possible que l'usuari executi les seves pròpies consultes SQL directament. En el model de dades de Django, una classe representa un registre d'una taula en la base de dades i les instàncies d'aquesta seran les files en la taula.

4.1.3.4 Proces d'una request

Tenint l'arquitectura en compte, es veurà a grans trets com es processa un request.

Quan Django rep una request, el primer que es fa és crear un objecte `HttpRequest` que la representa i servirà com a abstracció per treballar sobre diferents servidors.

Després es realitza la resolució de la URL. Això consisteix a seleccionar la funció de la view (a partir de la URL especificada en la request) que participará en la creació de la response.

Una vegada que hem resolt quina funció resoldrà la URL especificada, s'invoca a la funció de la view amb l'objecte request com a primer paràmetre. Aquí es realitza el treball pesat com: consultes a la BDD, càrrega de templates i generació d'HTML. Es retorna un objecte `HTTP response` o una excepció.

4.1.4 Base de dades

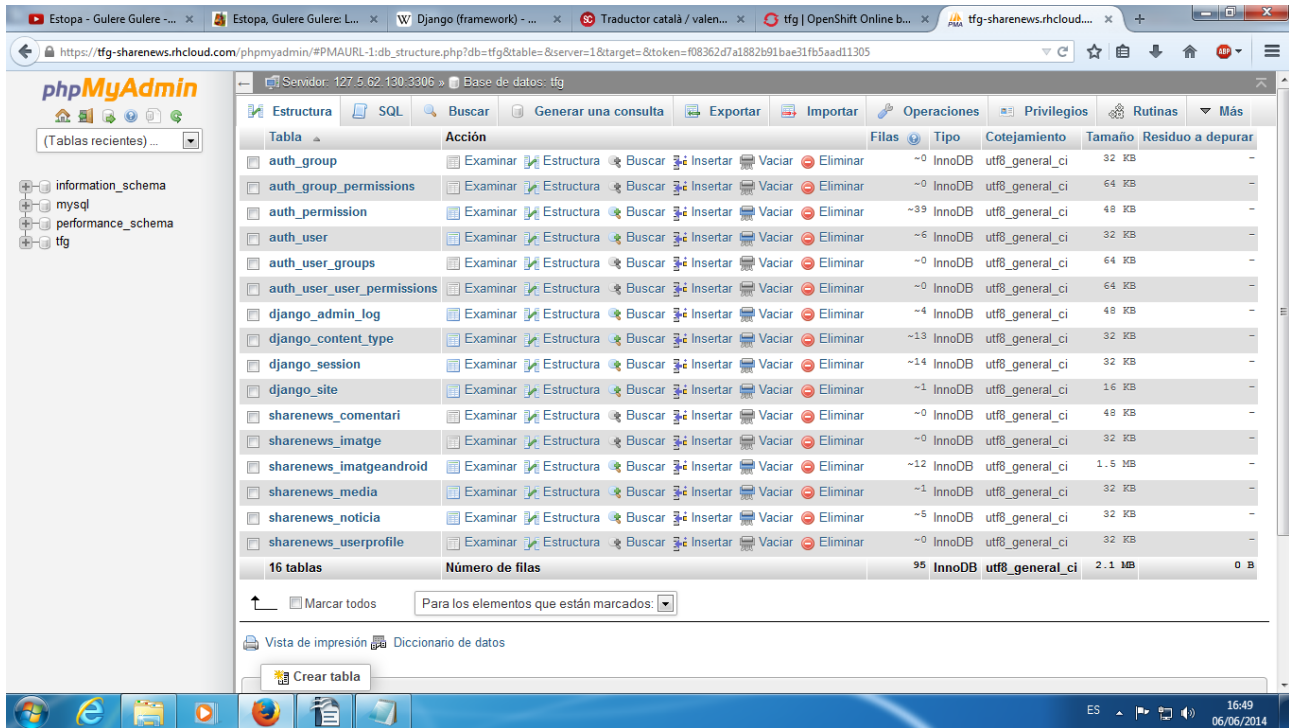
Tota la informació de les notícies, les seves característiques, usuaris, etc. han d'estar guardades a la base de dades. Aquesta serà utilitzada tan per la part web com per la part de Android. El llenguatge que s'ha utilitzat és el `MySQL` i les consultes que fem no són gaire complicades. El desenvolupador pot veure la informació que té o modificar-la a partir de la secció de “`phpMyAdmin`” el qual pot accedir des de `OpenShift`. Tot i així, com he mencionat avanç també es pot fer des de la consola però lògicament és més còmode amb la interfície.

Tots els nous objectes que s'afegeixen a la base de dades se'ls hi atorga un identificador únic, aquest serà la clau primària en totes les classes, apart Django atorga aquest identificador automàticament amb lo qual el desenvolupador o programador no s'ha de preocupar d'aquest. Apart les contrasenyes queden codificades per a més seguretat i inclús l'administrador no les pot veure, Django ja s'encarrega de fer-ho automàticament. També es pot donar la opció de que alguns atributs s'afegeixin automàticament com la Data (dia-hora-min-segons) o que si l'atribut en el moment que es crea l'objecte és buit per defecte posi un automàticament (és la mateixa idea que els constructors dels altres llenguatges de programació)

Django a més proporciona unes taules automàticament pel que fa relació de permisos d'usuaris, de administrador, de sessions, etc. Per suposat l'usuari pot afegir noves taules o modificar les existents però les mencionades anteriorment ja estan preparades per a que l'usuari des de les view pugui accedir-hi fàcilment i aplicar funcions de llibreries fetes ja per Django.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

A continuació (Figura 11. Taules Base Dades) es mostra una imatge de com es veuen les taules de la base de dades. Pot donar una idea al lector de com es la visió d'aquesta des de el lloc del desenvolupador.



The screenshot shows the phpMyAdmin web interface for a MySQL database named 'tfg'. The left sidebar lists the databases: 'information_schema', 'mysql', 'performance_schema', and 'tfg'. The main panel displays the 'Estructura' (Structure) tab for the 'tfg' database. It shows a list of 16 tables with their respective actions (Examinar, Estructura, Buscar, Insertar, Vaciar, Eliminar). The tables are:

Tabla	Acción	Filas	Tipo	Cotejamiento	Tamaño	Residuo a depurar
auth_group	Examinar Estructura Buscar Insertar Vaciar Eliminar	~0	InnoDB	utf8_general_ci	32 KB	-
auth_group_permissions	Examinar Estructura Buscar Insertar Vaciar Eliminar	~0	InnoDB	utf8_general_ci	64 KB	-
auth_permission	Examinar Estructura Buscar Insertar Vaciar Eliminar	~39	InnoDB	utf8_general_ci	48 KB	-
auth_user	Examinar Estructura Buscar Insertar Vaciar Eliminar	~6	InnoDB	utf8_general_ci	32 KB	-
auth_user_groups	Examinar Estructura Buscar Insertar Vaciar Eliminar	~0	InnoDB	utf8_general_ci	64 KB	-
auth_user_user_permissions	Examinar Estructura Buscar Insertar Vaciar Eliminar	~0	InnoDB	utf8_general_ci	64 KB	-
django_admin_log	Examinar Estructura Buscar Insertar Vaciar Eliminar	~4	InnoDB	utf8_general_ci	48 KB	-
django_content_type	Examinar Estructura Buscar Insertar Vaciar Eliminar	~13	InnoDB	utf8_general_ci	32 KB	-
django_session	Examinar Estructura Buscar Insertar Vaciar Eliminar	~14	InnoDB	utf8_general_ci	32 KB	-
django_site	Examinar Estructura Buscar Insertar Vaciar Eliminar	~1	InnoDB	utf8_general_ci	16 KB	-
sharenews_comentari	Examinar Estructura Buscar Insertar Vaciar Eliminar	~0	InnoDB	utf8_general_ci	48 KB	-
sharenews_imatge	Examinar Estructura Buscar Insertar Vaciar Eliminar	~0	InnoDB	utf8_general_ci	32 KB	-
sharenews_imatgeandroid	Examinar Estructura Buscar Insertar Vaciar Eliminar	~12	InnoDB	utf8_general_ci	1.5 MB	-
sharenews_media	Examinar Estructura Buscar Insertar Vaciar Eliminar	~1	InnoDB	utf8_general_ci	32 KB	-
sharenews_noticia	Examinar Estructura Buscar Insertar Vaciar Eliminar	~5	InnoDB	utf8_general_ci	32 KB	-
sharenews_userprofile	Examinar Estructura Buscar Insertar Vaciar Eliminar	~0	InnoDB	utf8_general_ci	32 KB	-
16 tablas	Número de filas	95	InnoDB	utf8_general_ci	2.1 MB	0 B

Below the table, there are options to 'Marcar todos' (Mark all) and 'Para los elementos que están marcados:' (For the elements that are marked:). At the bottom, there are links for 'Vista de impresión' (Print view) and 'Diccionario de datos' (Data dictionary), and a 'Crear tabla' (Create table) button.

Figura 11. Taules Base Dades

4.1.4.1 Diagrama ER

En la següent imatge (Figura 12. Diagrama ER) es pot veure la relació de les taules de la base de dades i les connexions entre elles.

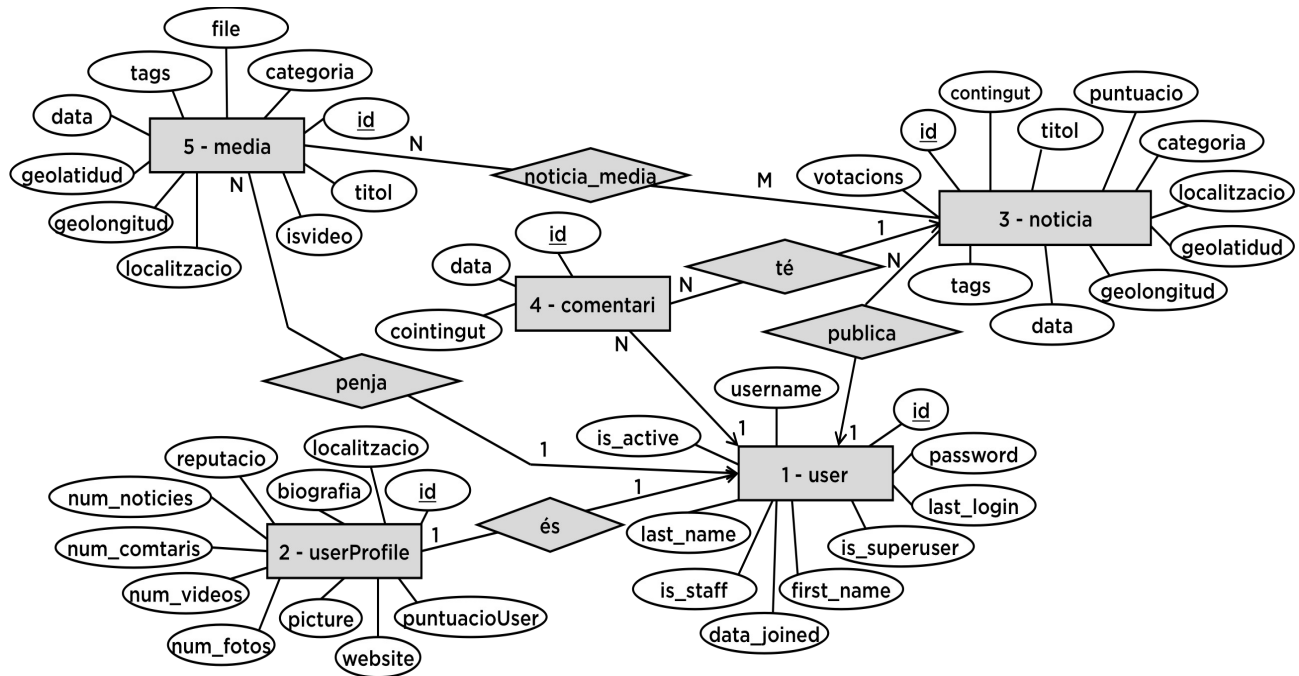


Figura 12. Diagrama ER

Definir que la punta de la fletxa significa en la relació 1 objecte i l'altre part N objectes. Les relacions N-M seran definides amb un romb el qual es la taula intermitja. Els atributs subratllats son els primaris.

4.1.4.2 Descripció Entitats de la taula

Tota la informació de la base de dades pot ser fàcilment accedida a través de les wiew's. Allà es on constitueix tota la lògica del projecte i totes les operacions que s'hagin de fer deixant així Android lliure d'executar el mínim d'operacions possibles. La informació que es torna sol ser en format json i/o File.

Les taules de la base de dades son comunes tan per la part web com per la part android.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

Entitat 1 - User

Aquesta taula ja ve creada per Django. Conte tota la informació relacionada amb un usuari.

- **id (INT (11))**: identificador usuari.
- **username (VARCHAR (30))**: nom d'usuari.
- **first-name (VARCHAR (30))**: nom real de l'usuari.
- **last-name (VARCHAR (30))**: cognom de l'usuari.
- **email (VARCHAR (75))**: correu electrònic de l'usuari.
- **password (VARCHAR (128))**: contrasenya.
- **is_staff (TINYINT (1))**: indica si pot entrar al portal d'administració.
- **is_superuser (TINYINT (1))**: indica si te tots els permisos.
- **last_login (DATETIME)**: última connexió de l'usuari.
- **data_joined (DATETIME)**: data de registre a la web.

Entitat 2 – UserProfile

Aquesta taula conte tota la informació relacionada amb l'usuari. A diferencia de la taula “user” que esta definida ja per Django aquí hi ha diferents atributs que serveixen per les funcionalitats descrites i les que en un futur hi ha intenció de crear.

- **id (INT (11))**: identificador usuari.
- **user_id (INT (11))**: l'usuari al qual va associat el perfil.
- **biografia (LONGTEXT)**: descripció de l'usuari.
- **localitzacio (LONGTEXT)**: localització usuari.
- **website (VARCHAR (200))**: pàgina web personal de l'usuari.
- **picture (VARCHAR (100))**: imatge del perfil d'usuari.
- **reputacio (INT (11))**: puntuació de l'usuari donada pel sistema.
- **puntuacio_user (INT (11))**: puntuació donada pels altres usuaris
- **num_fotos (INT (11))**: número de fotos que ha penjat l'usuari.
- **num_comentaris (INT (11))**: número de comentaris que ha fet l'usuari.
- **num_videos (INT (11))**: número de vídeos que ha penjat l'usuari.
- **num_noticies (INT (11))**: número de notícies que ha redactat l'usuari.

Entitat 3 - Noticia

Conte tota la informació relacionada amb una noticia creada per l'usuari. Títol, contingut, tags, categoria son alguns dels atributs d'aquest. Les imatges i vídeos que siguin adjuntades a la noticia es queden relacionades amb un foreignkey el qual es el seu identificador. Cada noticia es única i escrita per un sol usuari. En un futur es te pensat fer que les noticies puguin ser escrites per diferents usuaris.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

- **id (INT (11))**: identificador de la notícia.
- **autor_id (INT (11))**: usuari que ha escrit la notícia.
- **data (DATETIME)**: data i hora creació notícia.
- **títol (LONGTEXT)**: títol de la notícia.
- **contingut (LONGTEXT)**: cos de la notícia.
- **tags (LONGTEXT)**: tags associats a la notícia.
- **categoria (LONGTEXT)**: categoria de la notícia.
- **localitzacio (LONGTEXT)**: localització de la notícia.
- **geolatitud (DOUBLE)**: coordenada de latitud de la notícia.
- **geolongitud (DOUBLE)**: coordenada de longitud de la notícia.
- **puntuacio (DOUBLE)**: puntuació mitjana de la notícia.
- **votacions (INT (11))**: número de puntuacions donades a la notícia, guardat per poder calcular la mitjana.

Entitat 4 - comentari

Conte tota la informació necessària per guardar un comentari que s'hagi escrit en una notícia.

- **id (INT (11))**: identificador del comentari.
- **autor_id (INT (11))**: usuari que ha escrit el comentari.
- **noticia_id (INT (11))**: notícia a la que pertany el comentari.
- **data (DATETIME)**: data i hora en la que es va crear el comentari.
- **contingut (LONGTEXT)**: cos del comentari.

Taula 5 - media

Conte la imatge o vídeo en qüestió, també el autor que ha penjat la imatge i la latitud i longitud. Aquest dos últims serveixen per la geolocalització que utilitzarem en el mapa Google que s'explicarà més endavant com funciona.

- **id (INT (11))**: identificador de cada imatge o vídeo.
- **autor_id (INT (11))**: l'usuari que ha penjat la imatge o vídeo.
- **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.
- **tags (LONGTEXT)**: tags associats a la imatge o vídeo.
- **categoria (LONGTEXT)**: categoria de la imatge o vídeo.
- **localitzacio (LONGTEXT)**: nom del lloc d'on és la imatge o vídeo.
- **geolatitud (DOUBLE)**: coordenada de latitud de la imatge o vídeo.
- **geolongitud (DOUBLE)**: coordenada de longitud de la imatge o vídeo.
- **isvideo (TINYINT (1))**: variable per indicar si es tracta d'un vídeo o d'una imatge.

4.2 Android

En aquesta part s'explicarà com ha estat estructurada la part del projecte d'Android. Aquesta esta constituïda per tres parts bàsiques.

La primera es la programació en si del llenguatge. S'explicarà molt per sobre quines classes he utilitzat nomes pel fet que el lector si algun dia vol fer un projecte que tingui la mateixa estructura sigui capaç de fer-ho i tingui ja una visió global de que ha d'utilitzar, amb el que mencioni ja serà capaç de buscar la informació correcte. El codi el pot veure al projecte penjat.

La segona serà la part que es connecta amb el servidor. No s'explicarà el codi fet, si el lector el vol veure pot fer-ho mirant el codi del projecte, es mes el fet d'explicar el concepte de com funciona la connexió d'android amb URL's o servidors externs.

La tercera i ultima part es la utilització de Google Maps i com configurar-ho correctament per al seu funcionament. Al igual que en els altres dos casos si es vol veure el codi pot fer-ho accedint al projecte.

Un aspecte ha tenir en compte es la utilització de l'espai de memòria en android. Per tal d'evitar aquest problema totes les imatges i vídeos de les notícies i d'usuaris no son guardades al dispositiu mòbil sinó que son visualitzades a traves del servidor amb llibreries com “Vitamio” i “Picasso”. D'aquesta manera no s'utilitza espai a menys que sigui totalment necessari.

4.2.1 Classes usades

4.2.1.1 Activity

Una Activity en una finestra que conté la interfície d'usuari de l'aplicació. Aquesta te una part lògica que es el “.java” i l'altre gràfica que es el “.xml”. L'objectiu principal d'una activity és interactuar amb l'usuari. Des del moment que una activity apareix en la pantalla fins al moment que s'oculta, pansa per diverses etapes, conegut com el cicle de vida d'una activity.

Cada Activity te unes funcions especificques que s'executen en funció l'estat en que es troba l'aplicació. En la imatge de continuació es mostra quin es aquest flux. (Figura 13. Cicle vida activity)

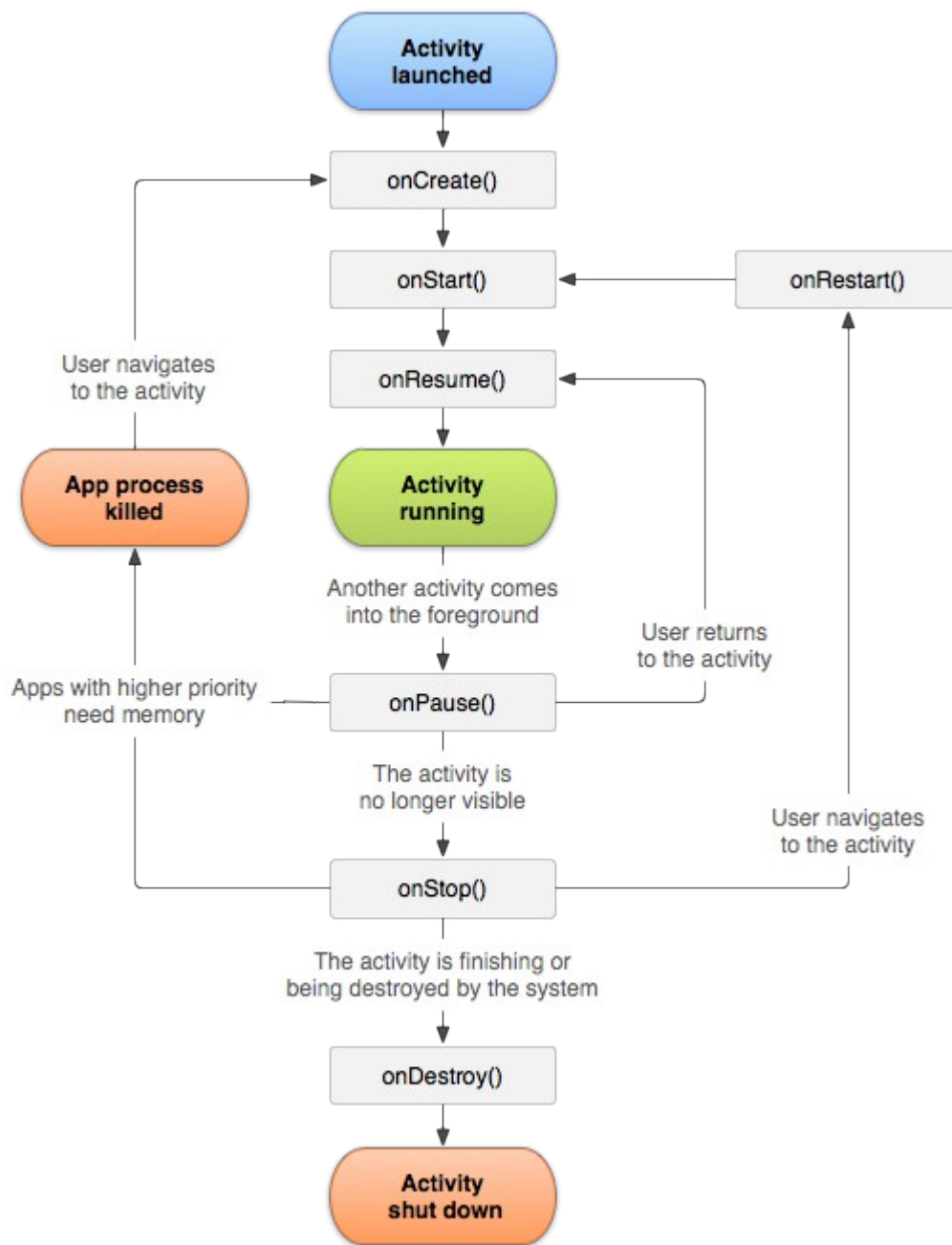


Figura 13. Cicle vida activity

4.2.1.2 Fragment

Un fragment es com una porció de la interfície d'usuari que pot afegir-se o eliminar-se d'una interfície de forma independent a la resta d'elements de l'activitat, i que per descomptat pot reutilitzar-se en altres activitats. Això, encara que en principi pot semblar alguna cosa trivial, permet dividir la interfície en diverses porcions de manera que puguem dissenyar diverses configuracions de pantalla, depenent de la seva grandària i orientació, sense haver de duplicar codi en cap moment, sinó tan sols utilitzant o no els diferents fragments per a cadascuna de les possibles configuracions.

4.2.1.3 Adapter

Un Adapter actua com a pont entre un AdapterView i les dades d'una Vista (View). L'adapter permet l'accés als elements de dades, aquest també és responsable de crear una vista per a cada element en la col·lecció de dades. Es pot dir, que l'adapter són col·leccions de dades, que assignem a una vista perquè aquesta els mostri, com per exemple en el cas de la meua aplicació que va mostrant les notícies que hi ha a la llista depenen la posició en que es trobi l'adapter. En resum, mitjançant l'ús d'adapters definim una forma comuna d'accedir a col·leccions de dades.

4.2.1.4 Runnable vs Thread

- Java no suporta herència múltiple, la qual cosa significa que només pot estendre una classe en Java així que una vegada estesa classe Thread no es pot estendre o heretar d'una altra classe en Java.
- La separació de tasques com a Runnable significa que podem tornar a utilitzar la tasca i també té la llibertat de cridar-les de diferents llocs. El Thread no es pot reiniciar un fil una vegada que es completa.

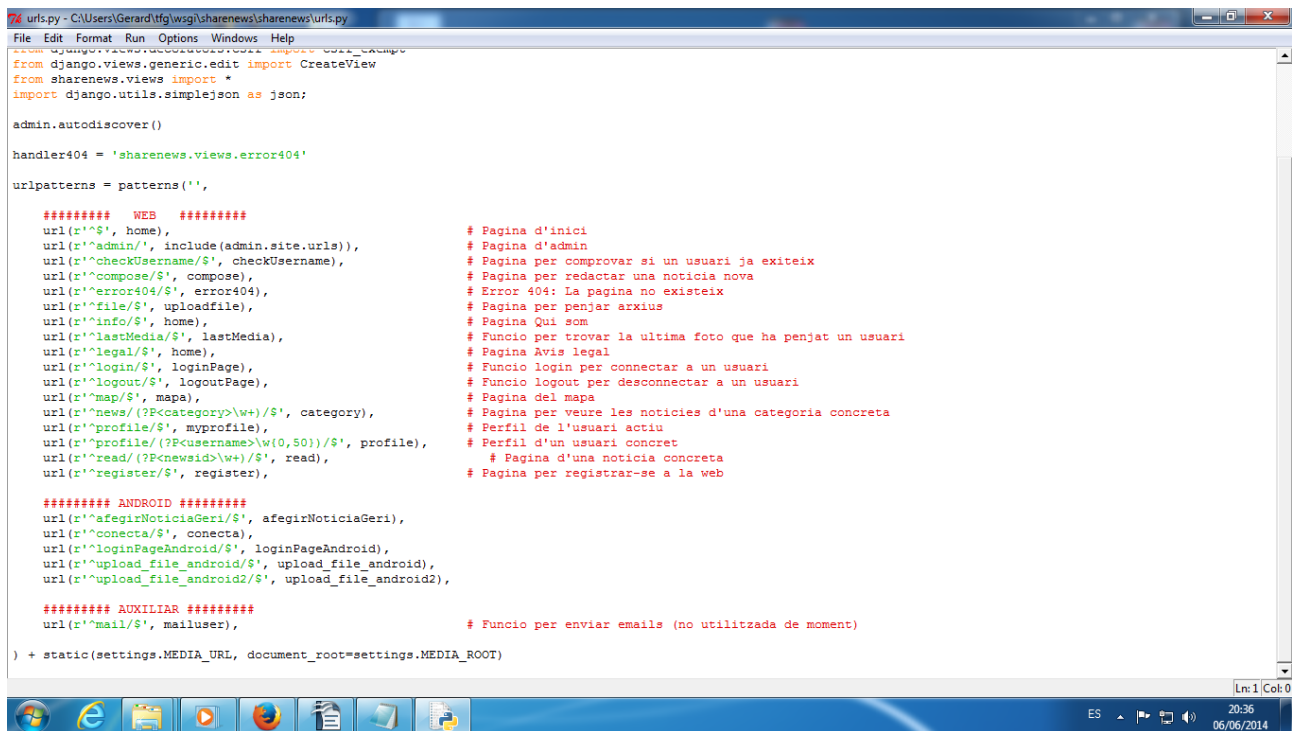
4.2.2 Connexió amb el servidor

L'aplicació ha d'estar en constant comunicació amb el servidor ja sigui perquè l'usuari ha escrit una notícia, vol veure una notícia o carregar notícies basant-se en un criteri de cerca.

El funcionament es molt senzill, des de Android s'envia una request en forma de "Post" a la url que s'especifica. Aquesta url (la qual formara part de la llista de direccions del servidor) serà processada per Django i depenen el contingut final de la url anirà a alguna funció o altre de la view.

Un exemple: des de android es connecta a la url "http://:Servidor/TFG/conectaURLescrita", Django al rebre la request llegeix la ultima part ("conectaURLescrita") i a l'arxiu urls.py (Figura 14. urls.py) es mirara si concorda amb alguna escrita allà, si concorda s'enviara la request a la funció corresponent que es trobi a views. Un cop finalitzada la lògica de la funció es retorna un resposta a android, i d'allà es segueix executa'n el codi d'android.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES



```
urls.py - C:\Users\Gerard\Documents\sharenews\sharenews\urls.py
File Edit Format Run Options Windows Help
from django.conf.urls.defaults import *
from django.views.generic.edit import CreateView
from sharenews.views import *
import django.utils.simplejson as json;

admin.autodiscover()

handler404 = 'sharenews.views.error404'

urlpatterns = patterns('',

    ##### WEB #####
    url(r'^$', home), # Pagina d'inici
    url(r'^admin/', include(admin.site.urls)), # Pagina d'admin
    url(r'^checkUsername/$', checkUsername), # Pagina per comprovar si un usuari ja existeix
    url(r'^compose/$', compose), # Pagina per redactar una noticia nova
    url(r'^error404/$', error404), # Error 404: La pagina no existeix
    url(r'^file/$', uploadfile), # Pagina per penjar arxius
    url(r'^info/$', home), # Pagina Qui som
    url(r'^lastMedia/$', lastMedia), # Funcio per trobar la ultima foto que ha penjat un usuari
    url(r'^legal/$', home), # Pagina Avis legal
    url(r'^login/$', loginPage), # Funcio login per connectar a un usuari
    url(r'^logout/$', logoutPage), # Funcio logout per desconnectar a un usuari
    url(r'^map/$', mapa), # Pagina del mapa
    url(r'^news/(?P<category>w+)/$', category), # Pagina per veure les noticies d'una categoria concreta
    url(r'^profile/$', myprofile), # Perfil de l'usuari actiu
    url(r'^profile/(?P<username>w{0,50})/$', profile), # Perfil d'un usuari concret
    url(r'^read/(?P<newsid>w+)/$', read), # Pagina d'una noticia concreta
    url(r'^register/$', register), # Pagina per registrar-se a la web

    ##### ANDROID #####
    url(r'^afegirNoticiaGerl/$', afegirNoticiaGerl),
    url(r'^connecta/$', connecta),
    url(r'^loginPageAndroid/$', loginPageAndroid),
    url(r'^upload_file_android/$', upload_file_android),
    url(r'^upload_file_android2/$', upload_file_android2),

    ##### AUXILIAR #####
    url(r'^mail/$', mailuser), # Funcio per enviar emails (no utilitzada de moment)

) + static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)
```

Figura 14. urls.py

Les request i responses poden portar informació adjuntada per el desenvolupador, en el cas del projecte per la part de android lo habitual es passar informació en format json i/o imatges i vídeos, tan a les anades com tornades. Un exemple: si l'usuari escriu una noticia nova l'aplicació android s'ha de connectar a la url corresponent i adjuntar a la request tota la informació de la noticia com el títol ,el contingut,l'usuari, imatges,etc. Un cop acaba la lògica de la funció de la view es retorna per response un String dient si s'ha pogut afegir correctament o no.

4.2.2.1 Com passar informació per la request

Per desgracia passar informació d'android a un servidor no es tan fàcil com al que s'esta acostumat a fer de passar informació entre dos classes de java, ja que la request en si es un cadena de text. Llavors com es pot passar objectes a traves de la request? Hi ha varies maneres de fer-ho, totes elles tenen els seus pros i contres, però al final en el cas de la part de l'aplicació d'adroid s'ha decantat per fer-ho de dos maneres diferents, una es utilitzant el “json” i l'altre el “MultipartEntityBuilder”. La primera es utilitza per passar tota la informació que serien String com el títol, contingut, tags,etc. I la segona s'utilitza per passar les imatges i els vídeos.

Al començament del projecte es passaven les imatges en base64 dintre del json però això va ser una cosa que en poc temps es va haver de canviar. El problema que hi ha es que els objectes guardats en base64 ocupen 1.37 cops mes que en format original i a la llarga això hagués gastat molt d'espai de la base de dades innecessàriament. Un altre punt que va fer canviar aquesta metodologia va ser que

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

encara que es pogués passar les imatges per la request en base64 els vídeos no podien. Això es degut a que un vídeo lògicament ocupa molt més espai que una imatge i les Entitys de les request (que era per on s'adjuntava la informació de les notícies i les imatges en base64) tenen un espai limitat que no es pot superar. Al intentar passar un vídeo hi havia error d'espai de memòria, i la única manera de solucionar-ho era dividir el vídeo en N parts i a la view ajuntar-los de nou. Aquesta solució no era bona, ni era òptima, llavors va ser quan es van buscar més formes de passar arxius i es va trobar el `MultipartEntityBuilder` el qual em permetia passar arxius grans sense dividir-los.

4.2.2.2 Json

JSON és l'acrònim per Javascript Object Notation, i encara que el seu nom ho digui, no necessàriament parteix de Javascript, de fet és un estàndard basat en text pla per a l'intercanvi d'informació, per la qual cosa s'usa en molts sistemes que requereixen mostrar o enviar informació per ser interpretada per altres sistemes, l'avantatge de JSON al ser un format que és independent de qualsevol llenguatge de programació, és que els serveis que comparteixen informació per aquest mètode, no necessiten parlar el mateix idioma, és a dir, l'emissor pot ser Java i el receptor PHP, cada llenguatge té la seva pròpia llibreria per codificar i decodificar cadenes de JSON. En el cas de l'aplicació s'envia la informació des de Android i es rep a Django que és python, en els dos casos hem de fer els corresponents imports per utilitzar les llibreries.

El principi bàsic és amb parells atribut-valor, aquests han d'estar tancats entre claus { , } que és el que defineixen l'inici i la fi de l'objecte. Un exemple podria ser aquest:

```
{
"titol": "Manifestacio UB",
"contingut": "Tota l'explicacio de la noticia",
"tags": ["manifestacio", "revolta", "drets"],
"autor": "Manel",
"localitzacio": true
}
```

Json bàsicament es basa en això, és una forma simple i còmoda de passar informació d'una plataforma a una altra. Pot passar diferents tipus d'objecte com Strings, int, double, list, etc. A la documentació oficial de Json que hi ha penjada a internet es pot mirar quins objectes accepta. Un cop s'ha creat el objecte json es converteix a String i s'adjunta a la request, llavors a la view es decodifica el json i es poden agafar els valors.

4.2.2.3 MultipartEntityBuilder

Per poder utilitzar aquesta classe s'ha de baixar un conjunt de llibreries de connexió d'Apache. Permet enviar arxius per la request i altres objectes. El funcionament es molt senzill i el lector pot veure en el codi com funciona amb lo qual aquí no s'explicarà. Simplement es mencionen les llibreries necessàries per a poder-ho utilitzar (Figura 15. Llibreries Apache). Es poden descarregar des de el Apache i son lliures. Importar una llibreria es molt senzill hi ha varies maneres, una d'elles i la que s'ha utilitzat per les llibreries relacionades amb apache es simplement agafar el “.jar” i arrastrar-lo dins la carpeta “libs” del projecte. La carpeta “libs” no ve creada per defecte, ha de ser el desenvolupador qui la crei manualment.

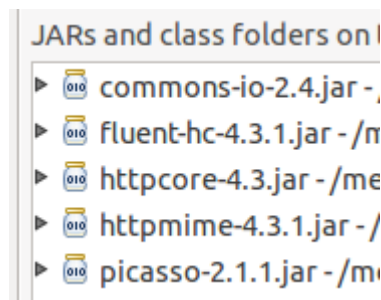


Figura 15. Llibreries Apache

4.2.3 Google Maps

En aquesta part s'explicaran les funcionalitats de google maps des de android. Per poder utilitzar totes les funcionalitats que aquest ofereix es requereix d'un seguit de condicions.

L'aplicació utilitza la API v2 per al projecte, aquesta es semblant a la v1 però afegeix algunes funcionalitats i classes. A mes la v1 quedara obsoleta d'aquí poc temps (si no ho esta ja). En la part de web s'utilitza una altre llibreria.

Per poder utilitzar google maps el desenvolupador ha d'aconseguir una API key, aquesta es donada per google un cop es registra l'aplicació al Google Play Services. Un cop es tingui la clau es col·loca al “manifest.xml”. Al final de la memòria es descriuen tots els passos a seguir per fer la configuració d'aquest.

Algunes de les funcionalitats que hi ha a l'aplicació mòbil i que estan relacionades amb el google maps son:

- Visualitzar mapa google: Permet visualitzar un mapa i moure's en ell, ampliar i reduir el zoom.
- Localització del dispositiu: Aquesta funció retorna dues coordenades, la latitud i la longitud. A mes si es desitja cada cop que canvia la localització d'aquest es pot actualitzar automàticament.
- Veure notícies en el mapa: Es mostra la localització de les notícies en el mapa.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

- Afegir localització a la notícia: Si l'usuari vol afegir la localització a la notícia s'agafen les coordenades i s'envien com a part de la informació de la notícia
- Canviar model del mapa: Es pot canviar la visualització d'aquest al gust de l'usuari, te varies opcions a elegir. Tot i així aquesta part no esta visible encara en l'aplicació perquè no esta del tot definit quines opcions han d'estar permeses per l'usuari amb el que fa la visualització del mapa.
- Canviar la posició d'inici del mapa: Mostra la part del mapa on es troba l'usuari inicialment.

5 Disseny

En aquest apartat s'explicara quines son les funcionalitats que te l'aplicació, el seu flux i el diagrama de classes.

5.1 Funcionalitats

1. L'usuari ha de poder veure les notícies globals.
2. L'usuari ha de poder llegir una noticia concreta.
3. L'usuari ha de poder llegir els comentaris d'una noticia.
4. L'usuari ha de poder comentar notícies.
5. L'usuari ha de poder crear notícies.
6. L'usuari ha de poder-se connectar
7. L'usuari ha de poder desconnectar de la seva sessió.
8. L'usuari ha de poder registrar-se (aquesta part redirigeix l'usuari a la web).
9. L'usuari ha de poder realitzar una cerca basant-se en la categoria.
10. L'usuari ha de poder realitzar una cerca basant-se en els tags.
11. L'usuari ha de poder visualitzar el mapa google per buscar notícies.
12. L'usuari ha de poder afegir imatges a les notícies que crea.
13. L'usuari ha de poder afegir vídeos a les notícies que crea.
14. L'usuari ha de poder afegir tags, categoria, etc. a les notícies que crea.
15. L'usuari ha de poder afegir la geolocalització a la noticia.
16. L'usuari ha de poder canviar foto perfil.

Requisits de l'aplicació

- Ha de poder-se connectar al servidor
- Ha de mostrar les notícies, comentaris, fotos i vídeos en forma de cascada utilitzant un “adapter”
- Ha de poder accedir a la SD per seleccionar les imatges i vídeos.
- Ha de poder utilitzar el GPS.
- Ha de ser capàs de crear thread's per les connexions amb el servidor.

5.2 Casos d'us

Tal com s'han organitzat les funcionalitats es poden diferenciar dos actor diferents, el primer es el “Usuari no connectat” i el segon el “Usuari connectat”. Cada un d'ells te les seves funcionalitats úniques i algunes compartides. (Figura 16. Casos d'us)

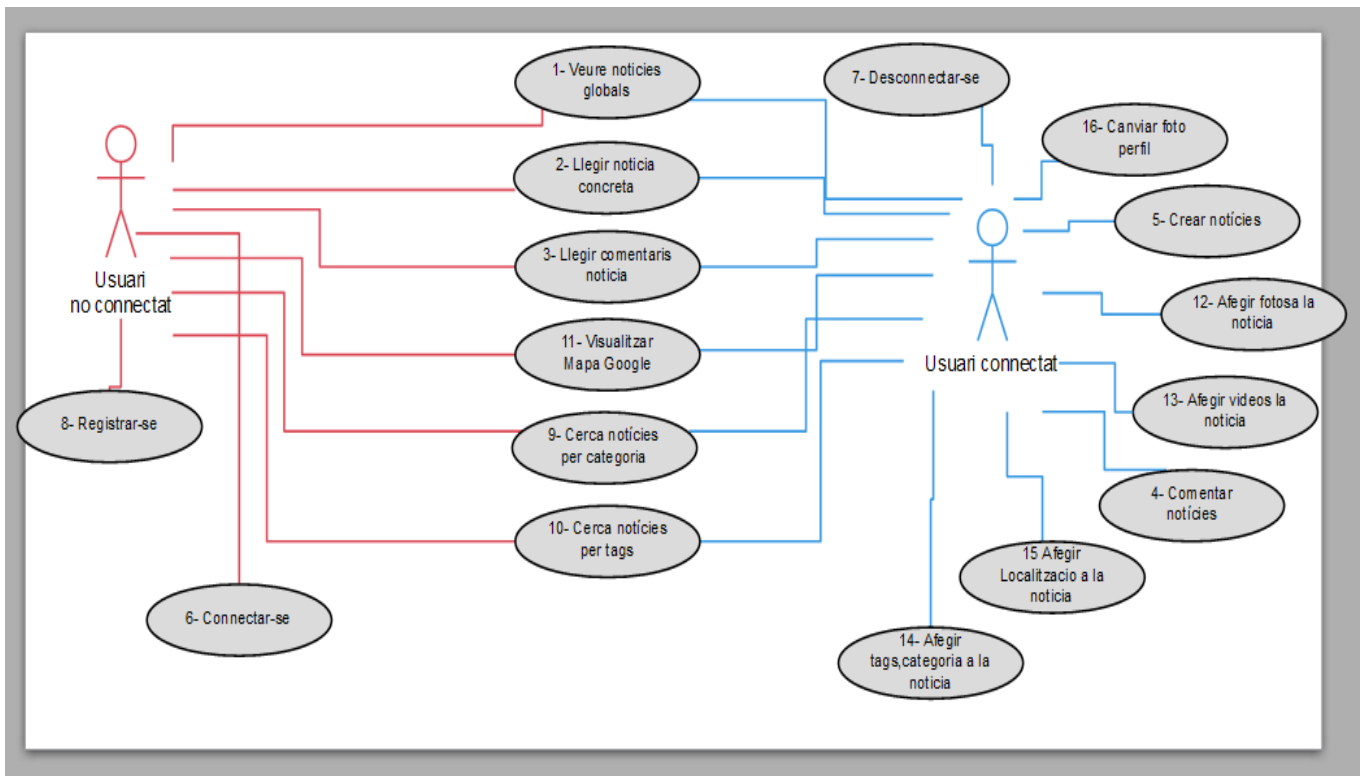


Figura 16. Casos d'us

5.3 Casos d'us textuals

Els casos d'us textuals serveixen per explicar d'una manera més natural el flux de cada una de les funcionalitats que l'usuari pot fer.

1 - Veure Notícies Globals

Cas d'us: Veure Notícia
Descripció: Veure les notícies globals en la pantalla d'inici de l'aplicació. No estan seleccionades basant-se en cap característica. Es mostrara una llista i a mesura que l'usuari vagi desplaçant la llista cap abaix s'aniran afegint mes des de el servidor.
Actor: Usuari connectat
Precondicions: Estar dins de l'aplicació i estar la finestra principal
Escenari principal: 1. L'usuari entra a l'aplicació des de el sistema operatiu android. 2. S'accedeix a la finestra on es mostren la llista de notícies del servidor.
Post condicions: L'aplicació detecta automàticament que aquelles X notícies que s'han mostrat no s'han de carregar de nou del servidor

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

2 - Llegir Noticia Concreta

Cas d'us: Llegir Noticia Concreta
Descripció: Seleccionar una noticia de la llista i poder-la llegir. Al seleccionar una noticia s'obrirà una nova finestra on es podrà llegir el contingut i totes les característiques que formen aquesta.
Actor: Usuari
Precondicions: Estar a la finestra principal de notícies globals
Flux Normal: 1.L'usuari selecciona una noticia concreta. 2.Sobre una nova finestra amb tota la informació d'aquesta.
Post condicions: Obrir una finestra nova amb la noticia seleccionada

3 - Llegir comentaris d'una noticia.

Cas d'us: Llegir comentaris d'una noticia.
Descripció: S'ha de poder llegir els comentaris que han fet altres usuaris a la noticia que s'estava llegint. Es mostraran en forma de cascada amb el mateix funcionament que es mostren les notícies.
Actor: Usuari no connectat i Usuari connectat.
Precondicions: Ser un usuari i estar llegint una noticia.
Flux Normal: 1. Des de la descripció d'una noticia que s'esta llegint, desplaçar-se a la dreta. 2. S'obre una finestra amb tot els comentaris. 3. Desplaçar cap abaix els comentaris per mostrar mes.
Post condicions: La noticia es penja al servidor i ja pot ser vista per tothom

4 - Comentar noticia

Cas d'us: Comentar Noticia.
Descripció: Un cop llegida una noticia s'ha de poder comentar-la. Un cop escrit el comentari i publicat serà afegit al servidor i qualsevol usuari podrà veure el comentari escrit en la noticia seleccionada.
Actor: Usuari connectat.
Precondicions: Estar connectat i llegint una noticia.
Flux Normal: 1. Des de la descripció d'una noticia que s'esta llegint, desplaçar-se a la dreta. 2. S'obre una finestra amb els comentaris de la noticia. 3. Escriu el comentari i clica el boto publicar. 4. S'afegeix el comentari al servidor.
Post condicions: el comentari es penja al servidor i ja pot ser visualitzada per tothom.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

5 - Crear noticia

Cas d'us: Crear Noticia.
Descripció: Quan l'usuari desitgi ha de ser capàs d'escriure una noticia i publicar-la. Aquesta ha de contindrà tota la informació desitja per l'usuari i un cop acabat serà pujada i emmagatzemada al servidor per ser visualitzada per a qualsevol usuari des de l'aplicació android o la web.
Actor: Usuari.
Precondicions: Estar loguejat.
Flux Normal: 1. Des de la finestra del perfil d'usuari de l'aplicació clica al boto de crear noticia. 2. Sobre una nova finestra. 3. Omple els camps necessaris i els voluntaris que cregui. 4. Clica el boto publicar.
Post condicions: La noticia es penja al servidor i ja pot ser vista per tothom

6 - Connectar-se

Cas d'us: Connectar-se
Descripció: Consisteix en que l'usuari en connecti amb la seva conta d'usuari. Ha d'estar ja registrat i que la conta no estigui caducada. Un cop connectat podrà realitzar accions com escriure notícies o comentar entre altres.
Actor: Usuari no connectat.
Precondicions: Estar dintre l'aplicació i no estar loguejat.
Flux Normal: 1. Des de la finestra principal de l'aplicació clica el boto de loguejar-se 2. Omple els camps de usuari i contrasenya 3. Envia les dades al servidor. 4. Es connectat correctament.
Flux Alternatiu: 1. El servidor retorna una senyal avisant que l'usuari o la contrasenya son incorrectes.
Post condicions: L'usuari queda connectat si ha omplert els camps correctes.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

7 - Desconnectar-se

Cas d'us: Desconnectar-se
Descripció: Consisteix en que l'usuari es desconnecti de la seva conta. Primer ha d'estar connectat. Un cop desconnectat no podrà escriure notícies ni comentaris entre altres. El perfil d'usuari es buidara.
Actor: Usuari connectat.
Precondicions: Estar connectat.
Flux Normal: 1. L'usuari entra a la finestra perfil usuari 2. Clica el boto en forma de “X” que hi ha a la dreta del nom d'usuari. 3. L'usuari queda desconnectat.
Flux Alternatiu: 1. No hi ha connexió a internet. 2. No es pot desconnectar.
Post condicions: L'usuari queda desconnectat.

8 -Registrar-se

Cas d'us: Registrar-se
Descripció: Un nou usuari vol crear-se una conta a l'aplicació. D'aquesta manera podrà utilitzar algunes de les funcionalitats que només es permeten als usuaris registrats i connectats.
Actor: Usuari no connectat
Precondicions: Ser un usuari no connectat i no tenir ja una conta.
Flux Normal: 1. L'usuari entra a la finestra de connectar-se 2. Clica al link que posa “Registrar-se” 3. Sobre l'explorador amb la finestra de la web de registrar-se 4. Omple tots els camps i envia la sol·licitud.
Flux Alternatiu: 1. El nom de usuari ja està utilitzat o el correu electrònic.
Post condicions: Queda l'usuari registrat i hi ha una nova conta a la base de dades.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

9 - Cerca per Categoria

Cas d'us: Cerca per Categoria
Descripció: Buscar notícies basant-se en categories. Les notícies poden incorporar a la descripció categories per definir la notícia.
Actor: Usuari no connectat
Precondicions: Haver clicat al boto de cerca personalitzada
Flux Normal: 1. Sobre la finestra de cerca personalitzada 2. Selecciona la categoria o categories que vol 3. Clicar al boto de buscar. 4. Es tanca automàticament la finestra de cerca. 5. La finestra principal mostra les notícies en qüestió.
Post condicions: La finestra principal mostra les notícies actualitzades.

10 - Cerca per Tags

Cas d'us: Cerca per Tags.
Descripció: Buscar notícies basant-se en uns criteris de cerca per tags. Les notícies penjades poden incorporar tags a la descripció d'aquest, aquesta cerca buscara totes les notícies que continguin els tags que l'usuari desitja.
Actor: Usuari no connectat
Precondicions: Haver clicat al boto de cerca personalitzada
Flux Normal: 1. Sobre la finestra de cerca personalitzada 2. Escriu els tags que desitja i els afegix a la llista. 3. Clicar al boto de buscar. 4. Es tanca automàticament la finestra de cerca. 5. La finestra principal mostra les notícies en qüestió.
Post condicions: La finestra principal mostra les notícies actualitzades.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

11 - Visualitzar Mapa Google

Cas d'us: Visualitzar Mapa Google
Descripció: L'usuari ha de poder obrir una finestra on es mostri el mapa google i la localització de les notícies.
Actor: Usuari connectat i Usuari Connectat
Precondicions: Estar dintre de l'aplicació
Flux Normal: 1. Des de la finestra principal on es mostren les notícies globals desplaçar-se a la dreta. 2. Sobre un nou fragment que es la visualització del mapa 3. Selecciona una noticia del mapa 4. Mostra la informació d'aquesta.
Flux Alternatiu: 1. Des de el mapa clica el boto d'endarrere del mòbil. 2. Torna a la pagina principal.
Post condicions: Visualització del mapa

12 - Afegir imatge a la noticia

Cas d'us: Afegir imatges a la noticia
Descripció: L'usuari afegeix una imatge a la noticia la qual esta creant i després serà publicada al servidor. Pot seleccionar mes d'una imatge. Aquestes es mostraran en una petita galeria per a que vegi quines ha afegit ja.
Actor: Usuari connectat
Precondicions: Estar connectat i esta dintre la finestra d'afegir noticia
Flux Normal: 1. Dintre de la finestra d'afegir noticia desplaçar-se a la dreta 2. Allà clicar el boto d'afegir imatge. 3. Sobre la galeria d'android. 4. Seleccionar la imatge desitjada 5. Es tanca la galeria i s'actualitza l'adapter de imatges. 6. Si l'usuari vol pot afegir mes imatges executant de nou el flux.
Flux Alternatiu: 1. L'usuari clica el boto de tornar endarrere del mòbil 2. No s'afegeix cap imatge.
Post condicions: S'ha afegit una nova imatges a la noticia

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

13 - Afegir vídeo a la notícia

Cas d'us: Afegir vídeo a la notícia
Descripció: L'usuari afegeix un vídeo a la notícia la qual esta creant i que sera publicada al servidor. Pot seleccionar mes d'una imatge. Aquestes es mostraran en una petita galeria per a que vegi quines ha afegit ja.
Actor: Usuari connectat
Precondicions: Estar connectat i esta dintre la finestra d'afegir notícia
Flux Normal: 1. Dintre de la finestra d'afegir notícia desplaçar-se a la dreta 2. Allà clicar el boto d'afegir vídeo. 3. Sobre la galeria d'android. 4. Seleccionar el vídeo desitjat. 5. Es tanca la galeria i s'actualitza l'adapter de vídeos. 6. Si l'usuari vol pot afegir mes vídeos executa'n de nou el flux.
Flux Alternatiu: 1. L'usuari clica el boto de tornar endarrere del mòbil 2. No s'afegeix cap imatge.
Post condicions: S'ha afegit una nova imatges a la notícia

14 - Afegir tags, categoria,etc. a la notícia

Cas d'us: Afegir tags, categoria,etc. a la notícia
Descripció: Quan es crea una notícia l'usuari ha de poder afegir informació addicional a la notícia, aquesta informació serveix per classificar-la i després per ser buscada en les cerques.
Actor: Usuari connectat i Usuari no connectat
Precondicions: Estar connectat i estar creant una notícia.
Flux Normal: 1. Des de la finestra d'escriure una notícia desplaçar-se a l'esquerra 2. Omplir els camps que cregui necessaris els quals poden ser seleccionar una categoria o varies i afegir un tag. 3. Per afegir un tag escriure'l al quadre de text i després clicar al boto "afegir tag"
Flux Alternatiu: 1. L'usuari desclica una categoria, aquesta es treu de la informació de la notícia.
Post condicions: S'ha carregat nova informació a la notícia.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

15 - Afegir geolocalització a la notícia

Cas d'us: Afegir geolocalització a la notícia
Descripció: L'usuari afegeix la localització a la notícia, aquesta es agafada per el GPS. Després alhora de buscar notícies per al mapa es mostrara
Actor: Usuari connectat
Precondicions: estar connectat i dintre la finestra escriure Notícia, tenir el GPS activat.
Flux Normal: 1. A la finestra principal d'escriure notícia desplaçar-se a l'esquerra 2. Seleccionar la opció d'afegir localització.
Flux Alternatiu: 1. No te activat el GPS, no es podrà afegir la localització quan es publiqui la notícia
Post condicions: Quan es publiqui la notícia s'afegira la localització.

16 - Canviar foto perfil

Cas d'us: Canviar foto perfil
Descripció: L'usuari canvia la foto de perfil. Selecciona una des de la galeria.
Actor: Usuari connectat
Precondicions: estar connectat i dintre la finestra escriure Perfil.
Flux Normal: 1. A la finestra de perfil d'usuari clicar el icona de canviar imatge 2. Seleccionar la imatge desitjada de la galeria
Post condicions: Quan es publiqui la notícia s'afegira la localització.

5.4 Patró MVC

El projecte en si es podria definir amb el patró “MVC” model, vista, controlador. A diferencia de Django el qual es definia amb un altre patró, es considera que el conjunt global de android i Django formen el patró que s'utilitza habitualment.

A continuació es mostra una figura que representa el patró (Figura 17. MVC). En la part web aquest serà diferent.

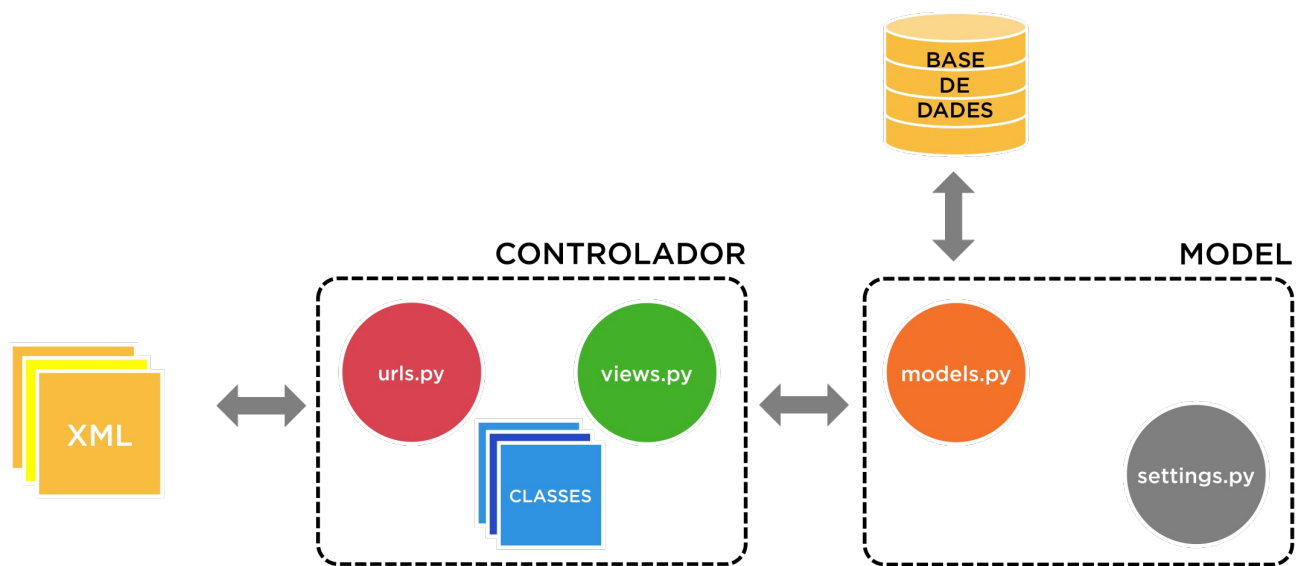


Figura 17. MVC

La vista, la imatge XML son tots els arxius “.xml” que es troben a android i formen la part visual de l'aplicació. Es tota la interfície pròpiament dit. L'única funcionalitat assignada es la part visual.

Al controlador s'engloben les classes d'android (els .java) i els dos arxius de Django “urls.py” i “views.py”. Aquests s'encarregaran de tota la lògica del programa i seran els responsables del seu correcte funcionament. Es on s'ha dut a terme la majoria del treball. Tal com s'ha mencionat avanç sempre que s'ha pogut s'ha deixat que les màximes operacions possibles fossin executades en el views.py ja que així es deixava el dispositiu mòbil lliure de realitzar operacions innecessàries que consumissin els recursos d'aquest.

A model hi ha el “models.py” i el “settings.py”. L'arxiu on s'ha treballat majoritàriament es en el models, ja que aquest era la definició de les taules de la base de dades . Aquests s'encarreguen de la connexió amb la base de dades pròpiament dita.

6. Desenvolupament

6.1 Programació

Primer s'explicara d'una manera resumida quines son les funcions mes rellevants que s'han implementat en la part de Django i exactament en l'arxiu “views”. Després s'explicara de la mateixa manera les classes de android que hi ha i quina funció desenvolupen.

6.1.1 Funcions Views

Al llarg del projecte s'han implementar un seguit de funcions necessàries per complir les funcionalitats establertes. Un detall a tenir en compte es que les funcions que son cridades des de android han de tenir “@csrf_exempt” com a override, sinó no poden ser executades.

- **loginPageAndroid(request):** Serveix per connectar-se un usuari a l'aplicació des de android. Retorna un valor, aquest valor des de android es llegira i informara al usuari si ha estat connectat correctament o hi ha hagut un error.
- **llegirNoticiesBasic(request):** Serveix per mostrar a l'usuari la llista de noticies globals que es mostren a la finestra principal de forma automàtica. No es basen en categories, ni en tags, simplement en l'ordre cronològic en que han estat creades. Retorna tota la informació d'una noticia, títol, contingut, tags, fotos, vídeos,etc.
- **llegirNoticiesCategoriaBasic(request):** Actua igual que la funció anterior però en aquest cas retorna les noticies depenen una categoria.
- **llegirNoticiesTagsBasic(request):** Actua igual que la funció de categories però en aquest cas retorna les noticies depenen els tags que s'han sol·licitat. Pot ser més d'un tag la cerca.
- **llegirComentarisBasic(request):** Retorna tots els comentaris que hi ha donada el identificador d'una noticia.
- **afegirComentari(request):** Afegeix un comentari a una noticia donada.
- **afegirNoticiaGeri(request):** Afegeix una nova noticia al servidor. Rep tota la informació per la request, crea una nova noticia i l'afegeix a la base de dades relacionada amb l'usuari que la creat, les fotos i els vídeos que s'han afegit.

6.2.2 Classes Android

A continuació s'explicarà d'una forma breu algunes de les classes més importants que es poden trobar a l'aplicació android. També es dirà si es un Activity, Fragment, etc. les extensions i implementacions que té la classe.

- **public class MainActivity extends FragmentActivity**

Conte l'adapter on es mostraran les finestres de perfil d'usuari i la llista de notícies amb la categoria a seleccionar.

- **public class PageAdapter extends FragmentPagerAdapter**

El adapter que conte les finestres mencionades anteriorment. Conte un arrayList de Fragments.

- **public class PerfilUsuari extends Fragment**

Conte tota la logica de la finestra de perfil d'usuari.

- **public class LlistaNotícies extends Fragment**

Conte les notícies carregades del servidor, independentment si aquestes han estat conseqüència d'una cerca o les globals.

- **public class Login extends Activity**

Conte tota la logica de la interfície del login. Crearà un thread quan es cliqui el boto connectar-se.

- **public class LlegirNoticia extends FragmentActivity**

Classe que es crida quan es vol llegir una noticia concreta. Conte un adapter el qual es format per un array de fragments els quals son:

- **public class LlegirNoticiaContingut extends Fragment**
- **public class LlegirNoticiaTags extends Fragment**

- **public class LlistaComentaris extends Fragment**

Conte tota la logica per llegir els comentaris d'una noticia seleccionada. Es un adapter de objectes Comentari.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

- public class EscriureNoticia extends FragmentActivity

Conte tota la logica per escriure una noticia. Conte un adapter el qual es formta per un array de fragments els quals son:

- **public class** EscriureNoticiaContingut **extends** Fragment
- **public class** EscriureNoticiaTags **extends** Fragment
- **public class** EscriureNoticiaMedia **extends** Fragment

- public class AfegirNoticiaWeb implements Runnable

Es un thread que es crea quan es vol afegir una noticia al servidor. La logica del thread es similar en tots els casos que es vol connectar al servidor com per exemple afegir un comentari, o comprovar el login. Nomes canvia en petites modificacions depenent el que s'envia i s'espera que retorni Django.

- public class Noticia

Conte tota la informacio d'una noticia.

- public class Comentari

Conte tota la informacio d'un comentari.

7. Resultats i test

Un cop finalitzat el desenvolupament s'inicia una avaluació dels resultats. En un principi son mes favorables del que s'esperava. Les funcionalitats que s'havien marcat funcionen, tot i això alguns petits detalls s'han de corregir.

Es va decidir realitzar uns testos a diferents usuaris reals. Aquests comprenien un ventall de diferents edats,professions,etc. Per assegurar que els resultats eren fiables i no es deixava ningun segment de la població al marge.

Els usuaris van haver de realitzar 2 tasques molt concretes, la primera escriure una noticia i la segona llegir una noticia i comentar-la.

7.1 Usuaris

Usuari 1

- **Escriure noticia:** 00:52 minuts
- **Llegir i comentar noticia:** 01:13 minuts
- **Obrir mapa google:** 00:12 minuts

El primer usuari es un estudiant d'enginyeria Electrònica. Te 23 anys i li agrada conèixer noies. Te un bon nivell en les noves tecnologies i navega sovint per internet. Esta familiaritzat amb les xarxes socials i sovint les utilitza.

S'ha guiat be per la aplicació i entenia les funcionalitats bàsiques, tot i aixi ha mencionat que no s'entenien alguns botons o no era capaç de relacionar-los. També ha mencionat algunes deficiències de l'aplicació i que la interfície era pobre.

Usuari 2

- **Escriure noticia:** 02:23 minuts
- **Llegir i comentar noticia:** 00:37 minuts
- **Obrir mapa google:** 00:05 minuts

El segon usuari es un home gran. Vesteix d'una manera tirada i amb barba. Un del seus hobbys es elaborar la seva pròpia cervesa artesanals.

Aquest usuari ha tardat ha passat bastant temps en la part d'escriure noticia. Ha redactat un text mes llarg del que els altres usuaris han fet i ha afegit bastants tags. També ha mencionat que trobava a faltar que alhora d'escriure no es pugui modificar la lletra ni afegir característiques com el “subratllat” o la “negreta”

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

Usuari 3

- **Escriure notícia:** 00:10 minuts
- **Llegir i comentar notícia:** 00:46 minuts
- **Obrir mapa google:** 00:15 minuts

El tercer usuari es un enginyer informàtic. Treballa en el desenvolupament de diferents aplicacions de jocs i en una empresa on s'encarrega de la base de dades. Posseïx una moto i juga molt a videojocs.

En el moment que li he demanat que crees una notícia i la publiques simplement ha publicat la notícia en blanc. L'aplicació ha seguit en curs normal perquè tal com s'ha configurat la base de dades quan els camps de la notícia estan buits per defecte posa un espai en blanc.

Usuari 4

- **Escriure notícia:** 03:54 minuts
- **Llegir i comentar notícia:** 04:34 minuts
- **Obrir mapa google:** 00:47 minuts

El quart usuari es una dona de la tercera edat, posseïx un molt bon nivell d'angles. Acostuma a anar a la piscina 2 o 3 cops a la setmana i li agradaria viatjar a Amèrica. No té experiència en les noves tecnologies, ni utilitza internet. El mòbil que té és molt antic, no és ni tàtil.

Ha tardat més que la mitja en realitzar les diferents accions, però tenint en compte la seva situació encara ho feia bé. Aprenia ràpid tot i no estar acostumada a aquestes tecnologies. Ha aconseguit realitzar les tasques tot i el temps que ha tardat. Un problema que s'ha trobat és que no aconseguia fer el símil de les icones amb les funcionalitats.

8. Estudi econòmic

En aquesta secció s'explicara com han estat distribuïdes les hores de feina al llarg del projecte i quines han estat les parts mes costoses de cada tasca.

El principal problema que hi ha hagut al llarg del projecte es que cada part nova que es volia afegir a l'aplicació s'havia de començar des de zero. No hi havia cap part en que s'hagi pogut reutilitzar els coneixements qua ja es tenien adquirits ja que cada funcionalitat de l'aplicació te una lògica molt diferent a les altres. El simple fet de per exemple escriure una noticia i pujar-la al servidor implicava que s'havia d'aprendre a agafar informació del servidor i tornar-la a android i allà descodificar-la i no simplement pujar la informació a al servidor des de android i ja esta.

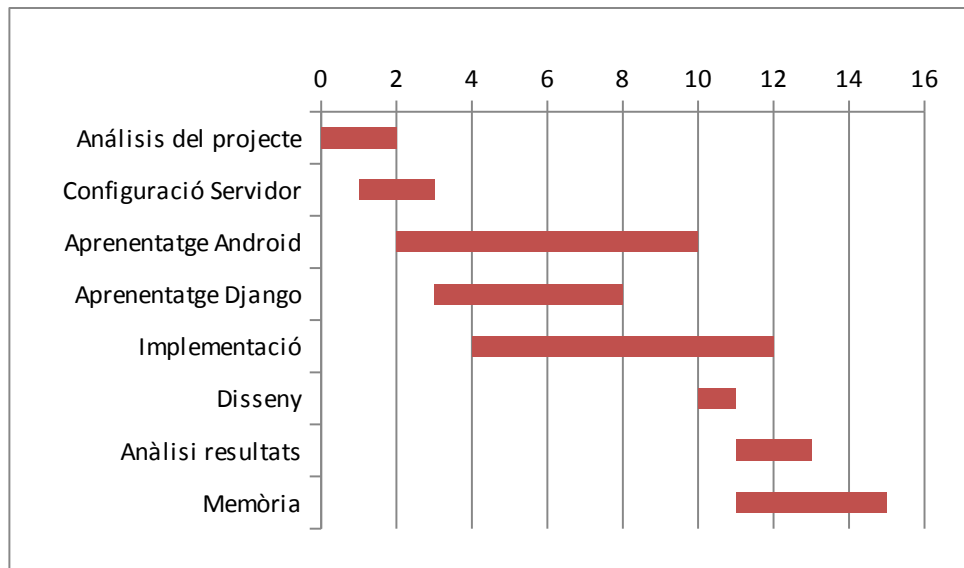
Potser el lector te la idea anticipada que tal com d'una banda transformes i passes la informació a l'altre banda serà el mateix però a la inversa, se li pot assegurar al lector que no es així. Cada funcionalitat que es creava s'havia d'aprendre-la a fer per partida doble, una per la part d'android i l'altre per la part de Django.

Per posar en relació hores suposem que en una funcionalitat nova se li ha invertit 10 hores, doncs d'aquestes 10 hores, 6 eren per aprendre quines funcions s'havien d'utilitzar, perquè i vigilar que fossin compatibles o no estiguessin obsoletes. Les altres 4 hores restants eren per fer l'implementació i ordenar tot. Un cop s'acabava la part d'android s'havia de repetir el mateix procediment per la part de Django.

En el principi del projecte s'havia planejat invertir un terç de les setmanes en la formació, però a la realitat contant totes les hores que es perdien en formar-se en cada nova funcionalitat es calcula que s'han perdut 2/3 parts del temps. (Figura 17. Diagrama Gantt)

Tot i que sigui un gast de temps important el punt a favor es que totes les hores de formació que s'han invertit fins ara son una bona base de coneixement per a l'estiu poder seguir treballant amb l'aplicació ja que la part difícil i costosa ja esta feta. Es podria descriure com una curva exponencial, al començament s'inverteixen moltes hores i sembla que no hi ha gaires canvis però cada cop s'aconsegueixen mes resultats en poc temps.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES



	Hores	Preu	Inversió
Anàlisi del projecte	10 h	13 €/h	130 €
Configuració de Servidor	5 h	12 €/h	60 €
Aprenentatge Android	100 h	12 €/h	1200 €
Aprenentatge Django	60 h	12 €/h	720 €
Estructura	40 h	15 €/h	600 €
Implementació	130h	13 €/h	1690 €
Disseny	15 h	13 €/h	195 €
Anàlisi resultats	2 h	11 €/h	22 €
Memòria	30 h	15 €/h	450 €
Hardware	0 h	0 €/h	0 €
Llicències	0 h	0 €/h	0 €
TOTAL	-	-	5067 €

9. Conclusions i treball futur

9.1 Conclusions

Després d'analitzar tota la feina que s'ha dut a terme al llarg del projecte considero que s'han assolit els objectius que s'havien proposat al inici d'aquest. Es cert que hi ha molt punts que m'hagués agradat millorat o treballar-los més però tenint en compte que era el primer cop en que treballava en un projecte així i a més propi considero que han estat uns bons resultats.

Al llarg del projecte s'han invertit moltes hores i esforç en el treball, he adquirit nous coneixements i he entès com funcionava el món de les xarxes socials i android. Encara que siguin moltes hores de feina, són tota la base necessària que permet després al desenvolupador crear una aplicació amb les funcionalitats i estil que prefereixi.

També mencionar, que al llarg del projecte hi ha hagut molts problemes trobats tan com per la part de configuració com la de implementació, i sobretot en Android.

9.2 Treball Futur

Tot i que al llarg del projecte no s'hagi pogut realitzar totes les funcionalitats que es tenien pensades ja que no hi havia temps per a tot s'espera poder afegir-les en algun moment. Algunes de les idees que es tenien pensades són aquestes:

La unió de diferents notícies en un mateix conjunt. El concepte seria el següent. Imaginem que hi ha varies persones que estan parlant del mateix esdeveniment, per exemple que hi ha una manifestació davant la UB, no te sentit que hi hagi creades N notícies diferents quan totes estan explicant el mateix, així doncs es donaria la possibilitat als usuaris de fer una unió d'aquestes i mostrar-les globalment. Seria com fer un “merge” del GitHub.

Una altra funcionalitat que es volia afegir però no ha donat temps es que des de android s'avisí al usuari quan s'ha comentat una notícia seva. Es simple de dur a terme i ofereix molta comoditat al usuari.

Per incitar a utilitzar l'aplicació s'havia pensat un sistema de puntuació per als escriptors. Aquests punts reflectiran la credibilitat o participació de l'usuari. A mesura que escriuen notícies i se'ls hi va puntuant aquestes positivament van guanyant punts. També es tindria en compte el nivell de l'usuari que puntua, no seria el mateix que un usuari nou posés bona qualificació a la notícia que un usuari ja experimentat i de alta puntuació ja que aquest últim seria més crític.

10. Referències bibliogràfiques

ListView Android

<http://jarroba.com/listview-o-listado-en-android/>

ViewPager Android

<http://developer.android.com/training/animation/screen-slide.html>

Activity Android

<http://developer.android.com/reference/android/app/Activity.html>

Fragments Android

<http://developer.android.com/guide/components/fragments.html>

Json

http://aiti.mit.edu/media/programs/indonesia-summer-2013/materials/json_django.pdf

Permisos internet Android

<http://stackoverflow.com/questions/8301944/cant-connect-to-local-host-form-android-emulator>

Post Json

<http://stackoverflow.com/questions/14119410/json-not-working-with-http-post-probably-around-setentity>

OpenShift

<https://openshift.redhat.com/app/login?then=%2Fapp%2Fconsole>

Get Json from response

<http://stackoverflow.com/questions/18073849/get-json-object-from-a-http-response>

Create Json

<http://stackoverflow.com/questions/2428092/creating-a-json-response-using-django-and-python>

Runnable vs Thread

<http://www.xyzws.com/Javafaq/what-is-the-difference-between-implementing-runnable-and-extending-thread/29>

Pass Image to Json

<http://mobile.cs.fsu.edu/converting-images-to-json-objects/>

Pick image from gallery Android

<http://viralpatel.net/blogs/pick-image-from-gallery-android-app/>

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

Open a URL in Android's web browser

<http://stackoverflow.com/questions/2201917/how-can-i-open-a-url-in-androids-web-browser-from-my-application>

Codigo colores HTML

<http://html-color-codes.info/codigos-de-colores-hexadecimales/>

Android : Upload large files using MultipartEntity

<http://stackoverflow.com/questions/23504191/android-upload-large-files-using-multipartentity>

Mapa Google

<https://developers.google.com/maps/documentation/android/v1/hello-mapview?hl=es>

Mapa Google Android

<http://www.sgoliver.net/blog/?p=3244>

Mapa Google v2

<http://www.androidhive.info/2013/08/android-working-with-google-maps-v2/>

POST a Multipart-Encoded File

<http://docs.python-requests.org/en/latest/user/quickstart/#post-a-multipart-encoded-file>

Galería de imágenes con el widget Gallery

<http://danielme.com/tip-android-25-galeria-de-imagenes-con-el-widget-gallery/>

Loading Remote Images in Android with UriImageView – Picasso libray

<http://www.donnfelker.com/loading-remote-images-in-android-with-uriimageview/>

Picasso

<http://square.github.io/picasso/>

how to play video from url

<http://stackoverflow.com/questions/2620049/how-to-play-video-from-url>

Get Current Location in Android

<http://javapapers.com/android/get-current-location-in-android/>

Estadísticas Android

<http://www.javahispano.org/android/2013/4/4/estadisticas-sobre-las-distintas-versiones-de-android.html>

Diccionario de Marketing

<http://www.foromarketing.com/glosario>

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

Casos d'us

http://www.codecompiling.net/files/slides/UML_clase_02_UML_casos_de_uso.pdf

Programación Android, startActivityForResult

<http://www.proyectosimio.com/programacion-android-startactivityforresult-lanzar-una-actividad-para-recibir-un-resultado/>

Django Advanced models

<http://www.proyectosimio.com/programacion-android-startactivityforresult-lanzar-una-actividad-para-recibir-un-resultado/>

Play Video Android

http://androidexample.com/Play_Video_File_-_Android_Example/index.php?view=article_discription&aid=124&aaid=144

How to add Rounded Corner borders to Android Layout

<http://tips.androidhive.info/2013/09/android-layout-rounded-corner-border/>

11. Annex

11.1 Manual d'usuari

Com registrar-se a l'aplicació?

Des de la finestra de perfil d'usuari (Figura f-1) clicar al boto (Figura b-1). S'obrirà la finestra login (Figura f-2) i allà simplement clicar al link registrar-se (Figura b-2). Aquest enllaç obrirà una nova pagina on es mostrara el formulari que hi ha a la web de registrar-se.



Figura b-1

[Registrar](#)

Figura b-2

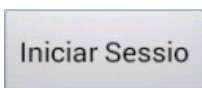


Figura b-3



Figura f-1

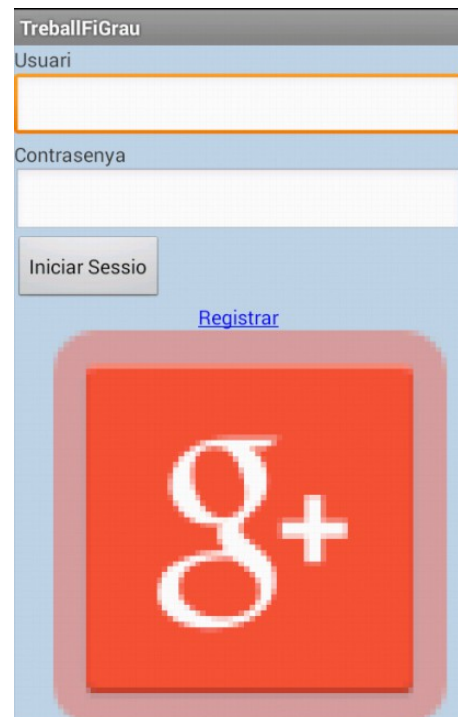


Figura f-2

Com connectar-se?

Des de la finestra login (Figura f-2) omplir els camps "Usuari" i "Contrasenya" i clicar boto "Iniciar Sessió" (Figura b-3). Si l'usuari i la contrasenya son correctes l'usuari quedara connectat , sinó es notificara amb un toast que es incorrecte. Per connectar-se ha d'estar registrat el usuari.

Un cop connectat es carregarà tota la informació de l'usuari i s'actualitzara la pàgina de perfil, incloent el seu nom i la imatge de perfil.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

Com redactar una nova noticia?

Des de la finestra de perfil d'usuari clicar el boto “Escriure Noticia”, s'obrirà un nou conjunt de finestres (Figura f-3, f-4, f-5) que es poden veure desplaçant la pantalla dreta i esquerra així accedir als diferents camps de la creació de noticia. Un cop l'usuari desitgi acabar la noticia ha de clicar al boto “Publicar” (Figura b-4).

Per afegir tags, l'usuari ha d'escriure el tag a la casella i després clicar al boto “Afegir Tag” (Figura b-5), aquest s'afegirà a llista de tags de la noticia. Per seleccionar una categoria o varies clicar a les corresponents. Per afegir la localització s'utilitza el mateix sistema que les categories.

Per afegir Fotos i Vídeos a la noticia s'han de clicar respectivament els botons “Afegir Foto” (Figura b-6) i “Afegir Vídeo” (Figura b-7). L'usuari pot afegir tants com vulguis. Les fotos i vídeos que l'usuari seleccioni es mostraran en un adapter.



Figura f-3

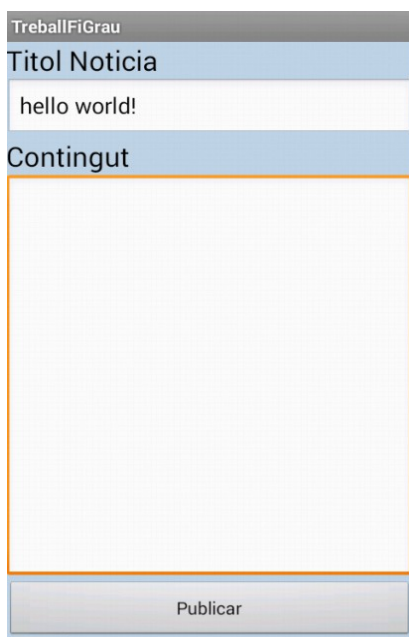


Figura f-4

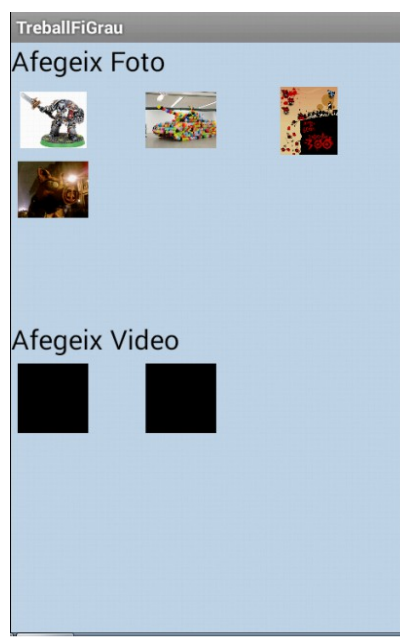


Figura f-5

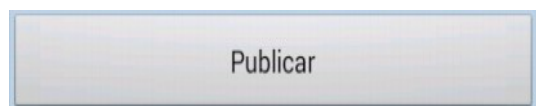


Figura b-4

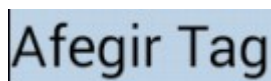


Figura b-5

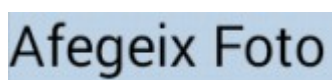


Figura b-6

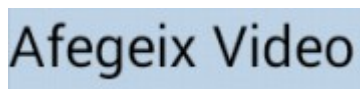


Figura b-7

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

Com veure notícies globals i per categoria?

Per veure les notícies globals accedir a la finestra de “Llista Notícies” (Figura f-6). Allà es mostren per defecte les notícies per ordre cronològic de creació, a mesura que l'usuari va desplaçant-se cap abaix de la pantalla s'afegeixen mes notícies descarregades des del servidor.

Si es volen veure les notícies globals un cop s'ha realitzar una cerca per tags o categoria clicar el boto (Figura b-8) de la finestra de perfil per visualitzar-les de nou.

Per veure les notícies relacionades amb una categoria clicar a la categoria desitjada (Figura b-9). Es mostraran a la finestra totes les notícies que tinguin com a característica la categoria seleccionada. Per veure mes categories de les que es mostren a la finestra desplaçar la barra dreta o esquerra.



Figura b-8

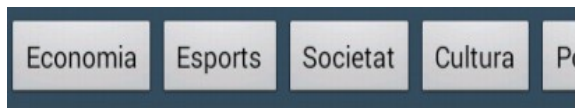


Figura b-9

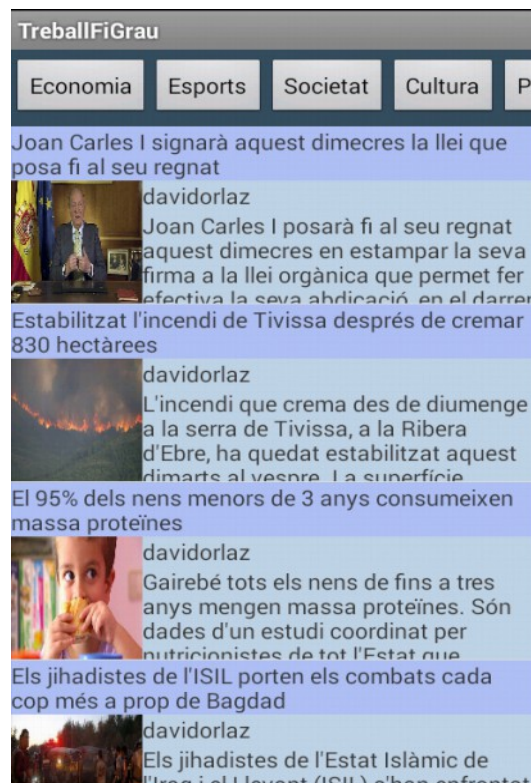


Figura f-6

Com llegir una notícia concreta i comentar-la?

Des de la finestra de “Llista notícies” seleccionar la notícia que es desitgi. Un cop seleccionada s'obrirà un conjunt de finestres (Figura f-7,f-8,f-9) les quals es poden veure desplaçant-se dreta i esquerra. Es el mateix funcionament que al escriure una noticia. Per veure les fotos i vídeos de la notícia clicar els botons fotos i vídeos respectivament (Figura b-10,b-11), això obrirà una galeria on es visualitzaran els mitjàs.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

Per llegir els comentaris d'una noticia desplaçar-se a la finestra “Llista Comentaris” (Figura f-9). Allà es mostraran tots els comentaris de la noticia els quals es descarregaran del servidor automàticament. Per veure mes comentaris desplaçar la llista cap abaix.

Per comentar la noticia l'usuari ha d'escriure el contingut en el quadre de text i en quan estigui clicar al boto “Publicar” (Figura b-12). Llavors s'afegira el comentari a la noticia.



Figura f-7



Figura b-10

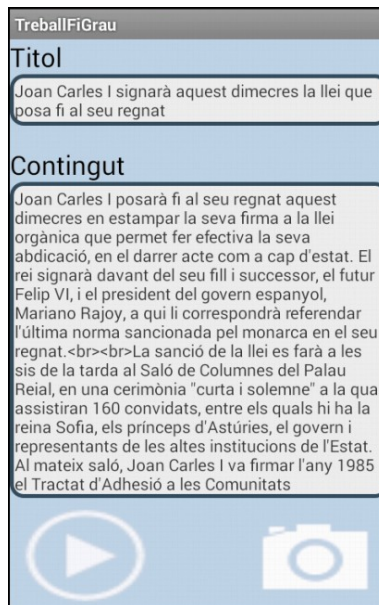


Figura f-8



Figura b-11

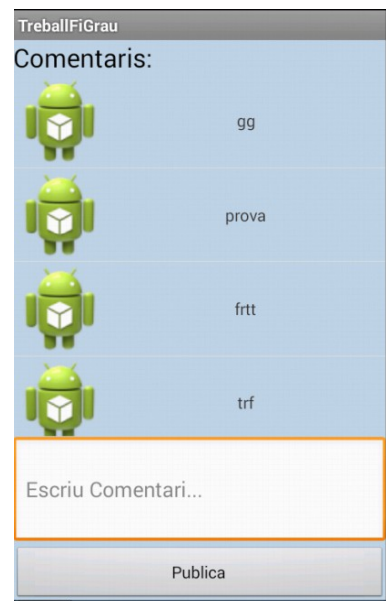


Figura f-9

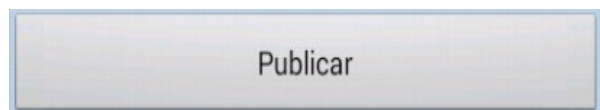


Figura b-12

Com visualitzar el Mapa Google i llegir notícies del mapa?

Des de la finestra del perfil d'usuari (Figura f-10) clicar al boto del mapa (Figura b-12). S'obrirà un mapa google (Figura f-11) en el qual l'usuari es podrà moure, fer zoom, i veure la localització de les notícies en forma de marker's. Al clicar a sobre d'aquest es mostrara el títol de la noticia.

Per llegir una noticia clicar sobre el títol del marker. S'obriran automàticament les finestres de llegir noticia amb la noticia seleccionada.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES



Figura f-10



Figura f-11

Com canviar la foto de perfil?

Des de la finestra de perfil clicar al boto (Figura b-13) de canviar de imatge. S'obrirà la galeria de fotos de android i l'usuari podrà elegir la imatge que desitgi. Un cop seleccionada es posarà la imatge a la finestra de perfil.



Figura b-13

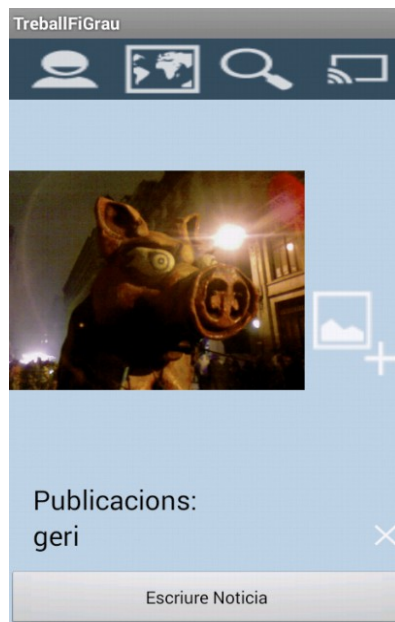


Figura f-12

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

Com desconnectar-se?

Per desconnectar-se de l'aplicació clicar al boto (Figura b-14) de la finestra perfil d'usuari. Això eliminara tota la informació del usuari guardada al mòbil.



Figura b-14

11.2 Manual d'instal·lació

11.2.1 Eclipse instal·lació

Accedir a la pagina web d'Eclipse. <http://www.eclipse.org/downloads/>

Un cop dins, descarregar l'arxiu (Figura e-1)



Figura e-1

La versió de 32-bits o 64-bits s'elegeix en relació el sistema operatiu de cada ordinador, s'ha de mirar quin suporta.

Un cop descarregat l'arxiu es descomprimeix i es mou la carpeta descomprimida a on desitgi l'usuari. La pròpia descompressió es la instal·lació d'eclipse.

Nota: Si no esta instal·lada la maquina virtual de Java no es podrà executar eclipse. Per instal·lar-lo nomes s'ha d'anar a <http://java.com/es/download/> i allà clicar a l'icona (Figura e-2)



Figura e-2

Un cop realitzat aquest dos passos ja nomes queda obrir eclipse i importar el projecte i les seves llibreries corresponents. Recordar importar totes les llibreries.

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

Maquina Virtual

La maquina virtual utilitzada te les següents característiques.

- **AVD Name:** 4.4.2_API_Google
- **Device:** Nexus One (3.7" 480x800 hdpi)
- **Target:** Google APIs – API leves 19
- **CPU/ABI:** ARM (armeabi-v7a)
- **Skin:** WVGA800
- **RAM:** 512
- **VM Heap:** 32
- **Internal Storage:** 1000 mb
- **SD Card:** 1000 mb

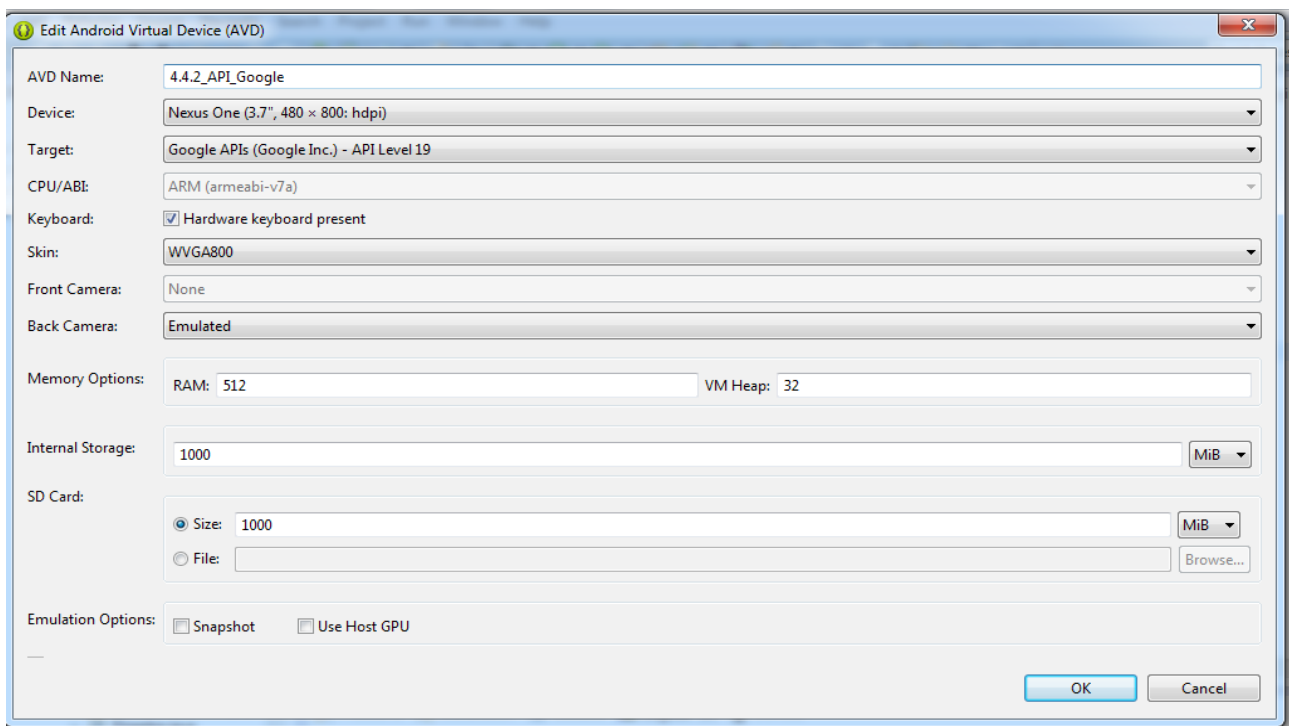


Figura e-3

Django

Per veure instal·lació de Django anar al TFG de “DISSENY D'UNA XARXA SOCIAL WEB PER A COMPARTIR NOTÍCIES”. No s'explica aquí ja que tècnicament per utilitzar l'aplicació android no cal configurar Django ni el servidor ja que ja estan configurats.

11.2.2 Google Maps Instal·lació

En aquesta part explicaré com poder utilitzar google maps des de android, faré un petit tutorial per a que el lector el dia que vulgui utilitzar aquesta API sigui capàs de fer-ho, potser sembla una mica llarg però puc assegurar que seguint els passos no tindrà cap problema i un cop configurat no haurà de tocar-ho mai més.

He utilitzat la API v2 per al projecte, aquesta es mes o menys com la v1 però afegeix algunes funcionalitats i classes, apart la v1 quedara obsoleta d'aquí poc temps (si no ho esta ja). Se que el meu company utilitza una altre llibreria ja feta especialment per web i crec que per versió google ja anava per la v3 o v4, simplement comentar-ho per si algun dia el lector duu a terme un projecte similar a aquest, que sigui conscient que son dos llibreries diferents.

Comentat tot això ja passem a la guia per poder utilitzar els mapes de Google a l'aplicació.

- Descarregar i configurar els Google Play Services
- Obtenir una API key
- Definir las especificacions en el Application Manifest
- Afegir un mapa a l'aplicació.

Una vegada ens hem baixat tot el SDK de Android, executarem el SDK Manager per descarregar-nos d'una banda una versió del SDK igual o superior a Android 4.2.2 i per l'altre costat, el paquet extra de Google Play Services. (Figura m-SDK) Tenir present que la meua aplicació android s'haurà utilitzat el 4.4.2.

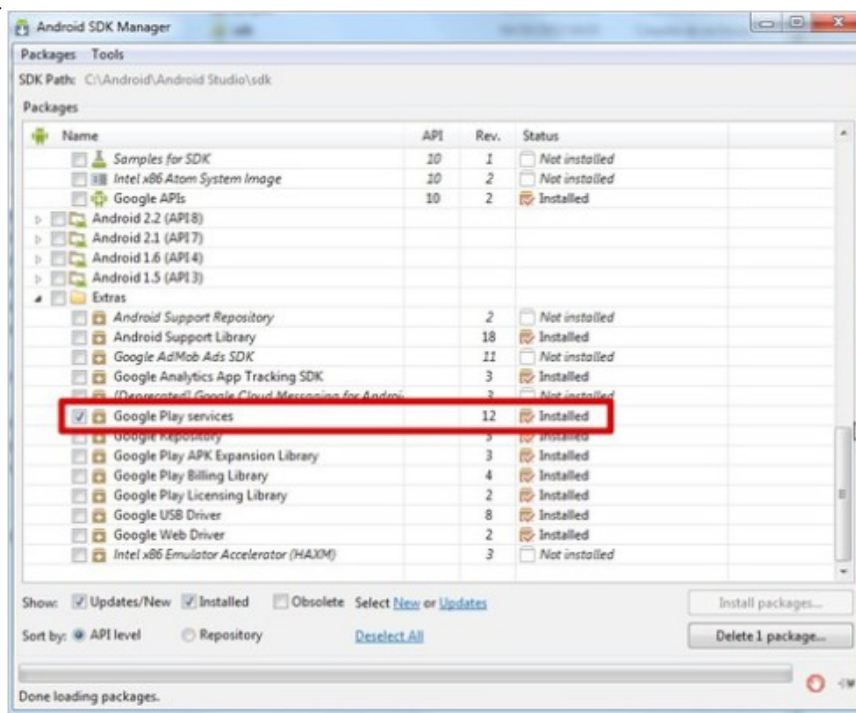


Figura m-SDK

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

Un cop descarregat es troba en la carpeta <android-sdk>/extras/google/google_play_services/. El següent pas serà carregar aquesta carpeta com a una llibreria així podrem utilitzar-la des de el nostre projecte. Per això, n'hi ha prou amb obrir l'Eclipse i prémer File -> Import -> Android -> Existing Android Code into Workspace i buscar la carpeta esmentada just a dalt. La ruta pot canviar depenen de la configuració de l'eclipse i el workspace però l'objectiu es aconseguir la carpeta de google_play_service.

Aconseguir la API key de Google Maps

Per accedir als servidors de Google Maps amb la API de Google Maps hem d'afegir una API key a la nostra aplicació. Aquesta clau estarà associada exclusivament a la nostra app gràcies als certificats que utilitza Android i funcionarà o no en funció d'això. Així doncs, necessitarem abans de res el codi SHA-1 de la nostra aplicació per poder sol·licitar la API key. Per saber quina tenim anem a Eclipse -> Window -> Preferences -> Android -> Build (Figura m-Codi SHA1)

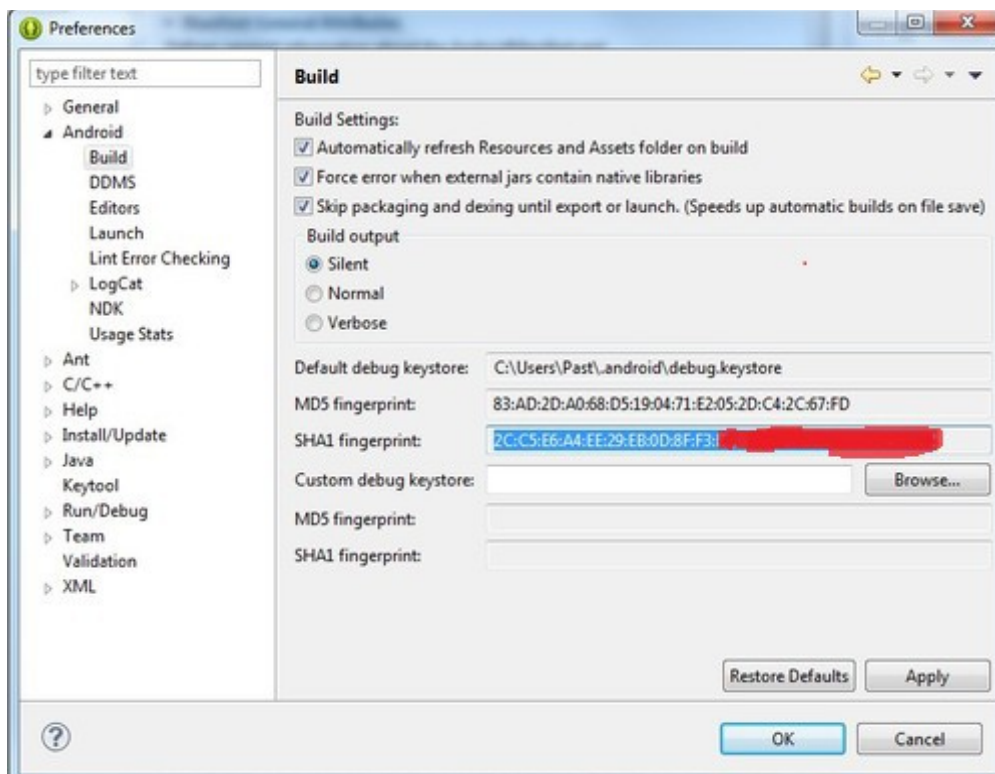


Figura m-Codi SHA1

A continuació accedim a la consola de Google API's. Un cop allà, anem a la opció APIS i activem (activar es simplement posant en ON el boto) “Google Maps Android API V2” (Figura m-Google Maps Android API V2)

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

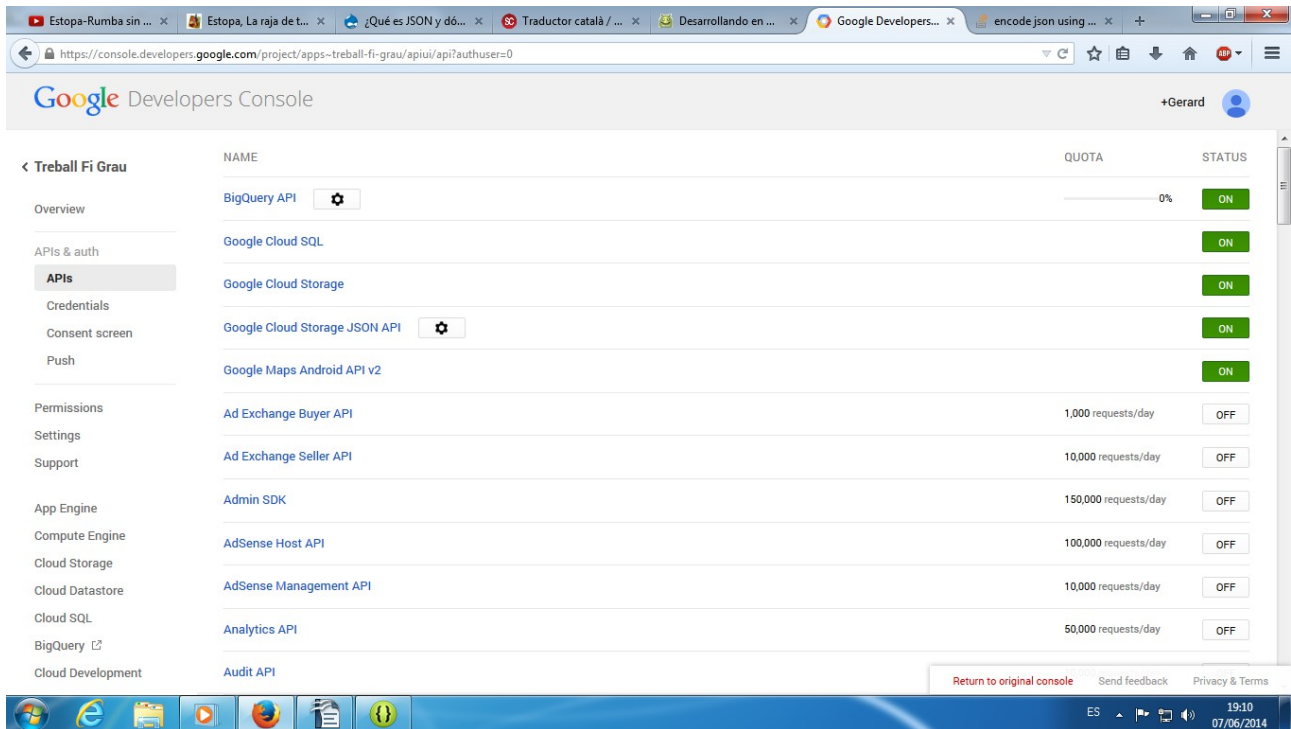


Figura m-Google Maps Android API V2

Un cop fet anem a “Credentials” i allà a la opció de generar una nova Key. Dintre de la opció haurem de col·locar el codi SHA1 del nostre projecte android juntament amb el nom del paquet d'aquest separat pel símbol punt i coma ' ; '.

Si tot ha funcionat correctament s'haurà de generar automàticament la Key que utilitzarem al projecte. Poso una imatge de com hauria de quedar.

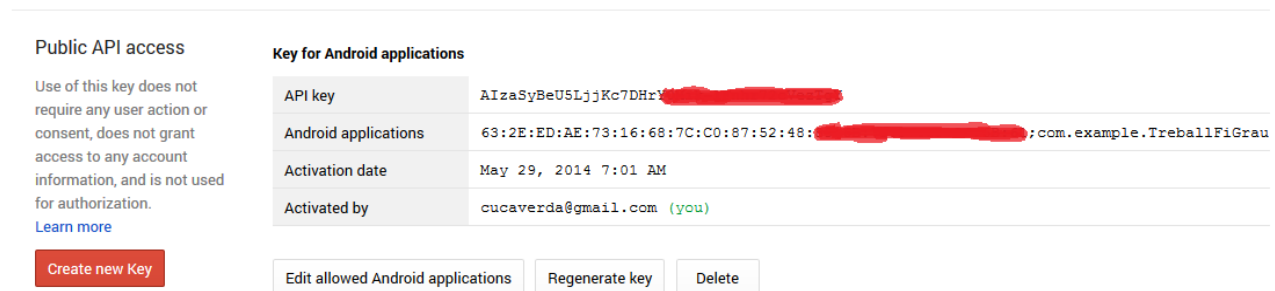


Figura N API Key

Un cop tenim la API key es moment de posar-la a l'aplicació. En el “AndroidManifest.xml” afegim el següent codi dintre de <application>

```
<meta-data
android:name="com.google.android.maps.v2.API_KEY"
android:value="API_KEY"/>
```

DISSENY D'UNA XARXA SOCIAL ANDROID PER COMPARTIR NOTÍCIES

Els següents permisos també s'han d'afegir, sense ells no permetríem a l'aplicació que funcione tot el que hem fet.

•**android.permission.INTERNET** Usat per la API per descarregar els mapes dels servidors.

•**android.permission.ACCESS_NETWORK_STATE** Permet a la API comprovar l'estat de la connexió per veure si pot descarregar dades.

•**com.google.android.providers.gsf.permission.READ_GSERVICES** Permet a la API accedir als serveis de Google basats en la web.

•**android.permission.WRITE_EXTERNAL_STORAGE** Permet a la API emmagatzemar en cache dades del mapa en la SD.

•**android.permission.ACCESS_COARSE_LOCATION** Permet a la API utilitzar la WiFi o les dades mòbils (o tots dos) per determinar la localització.

•**android.permission.ACCESS_FINE_LOCATION** Permet a la API usar el GPS per determinar la localització amb precisió.

Google Maps utilitza OpenGL ES v2 per renderitzar els mapes. Si no hi ha instal·lat OpenGL ES v2, el mapa no apareixerà, per tant també és recomanable afegir-ho.

```
<uses-feature  
android:glEsVersion="0x00020000"  
android:required="true"/>
```