



Cobertura de Xarxes d'Operadors Mòbils (agrupades per regió)

Memòria de Projecte Final de Màster

Màster Universitari d'Aplicacions Multimèdia

Itinerari professional

Universitat Oberta de Catalunya

Autor: Juan Luis Marí Pardo

Consultor: Sergio Schvarstein Liuboschetz

Professor: David García Solórzano, Laura Porta Simó

14 de gener 2018

Crèdits/Copyright

© *Juan Luis Marí Pardo*

Reservats tots els drets. Està prohibit la reproducció total o parcial d'aquesta obra per qualsevol mitjà o procediment, compresos la impressió, la reprografia, el microfilm, el tractament informàtic o qualsevol altre sistema, així com la distribució d'exemplars mitjançant lloguer i préstec, sense l'autorització escrita de l'autor o dels límits que autoritzi la Llei de Propietat Intel·lectual.

FITXA DEL TREBALL FINAL

Títol del treball:	<i>Cobertura de Xarxes d'Operadors Mòbils (agrupades per regió)</i>
Nom de l'autor:	<i>Juan Luis Marí Pardo</i>
Nom del consultor/a:	<i>Sergio Schvarstein Liuboschetz</i>
Nom del PRA:	<i>David García Solórzano, Laura Porta Simó</i>
Data de lliurament (mm/aaaa):	<i>01/2018</i>
Titulació o programa:	<i>Màster Universitari d'Aplicacions Multimèdia</i>
Àrea del Treball Final:	<i>Treball de Final de Màster Professionalitzador</i>
Idioma del treball:	<i>Català</i>
Paraules clau	<i>Escaneig de xarxes mòbils, mapeig de senyals basat en regió d'operadors mòbils, llenguatges Android – Django – Angular2</i>
Resum del Treball:	
<p>Actualment vivim a l'era de la informació, un temps caracteritzat pels grans avanços que s'han produït a nivell digital i tecnològic, especialment degut a la seva influència en les nostres vides. Així, estar comunicats i amb accés a la xarxa en tot moment s'ha convertit en una necessitat vital pel desenvolupament de moltes persones, que gràcies a dispositius com tablets i smartphones s'han aconseguit suplir, però que depèn en gran mesura de la cobertura de l'operador mòbil contractat. Concretament, aquest projecte sorgeix per tal d'oferir un ampli servei d'estadístiques i informació sobre quines són les senyals prioritàries segons l'àrea geogràfica interessada. Per realitzar-ho, s'han realitzat dues parts de desenvolupament ben diferenciades, una de les quals permet els usuaris conèixer i compartir a una base de dades les dades de senyals mòbils captades (aplicació Android), i la segona de les quals permet els usuaris visualitzar els resultats agrupats per regió de la intensitat de les totes senyals compartides, així com pels diferents tipus de senyals 2G, 3G, 4G (aplicació Web). Dels resultats obtinguts han sorgit diferents conclusions, com ara bé que les senyals 4G tot i ser les que ofereixen una millor velocitat de connexió reben generalment una pitjor cobertura que les anteriors generacions, o com ara bé descobrir els operadors que ofereixen millor cobertura a zones "mortes" (metro, muntanya) de connexió. D'aquesta manera, s'espera que aquest servei serveixi els operadors mòbils per millorar les seves infraestructures i els usuaris per tenir coneixement de les xarxes mòbils òptimes a qualsevol regió del mapa.</p>	

Abstract:

Nowadays, people live in the Information Age, a time characterized by the big advances at the digital and technological levels, especially due to their influence on our lives. Thus, being connected and having constant availability to the network has become a vital necessity for the daily lives of many persons, which have been able to accomplish thanks to devices such as tablets and smartphones, but that depends especially on coverage and intensity of the contracted mobile operator. Concretely, this project arises in order to offer a broad statistical and information service about which are the priority signals according to the concerned geographic area. In order to accomplish its purpose, two well-differentiated parts of development have been carried out, one of which allows users to know and share data on the captured mobile signals (Android application), and the second one that allows the users to view the results of all Shared signals grouped by region as well as by its different type of technology: 2G, 3G, 4G (Web application). The obtained results have derived in some conclusions, such as the fact that the 4G signals, although offering the fastest connection to the network, generally receive worse coverage than its previous generations, or as well as discovering the operators that offer better coverage under "death" areas (subway, mountain) of connection. This way, it is expectable that this service will be used by mobile operators to improve their infrastructures and also by users to get acquainted with the best mobile networks in any region of the map.

Dedicatòria

Aquest treball se'l dedico a la meva família, especialment als meus pares Juan José i Agustina, que sempre ha confiat en mi i que m'ha recolzat en tot moment per tal que pugui seguir endavant.

Agraïments

Agraeixo el meu consultor Sergio S. per les seves indicacions durant el transcurs d'aquest projecte. Agraeixo la meva empresa de treball (*Visual Engineering*) per la formació donada durant aquests més de dos anys treballant, sense la qual no hagués set capaç de realitzar aquest treball. Reconeixement especial per l'ajuda dels meus companys de treball i amics Marc M. i Néstor G., que m'han ajudat a resoldre aspectes tècnics necessaris pel desenvolupament d'aquest projecte.

Resum

Actualment vivim a l'era de la informació, un temps caracteritzat pels grans avanços que s'han produït a nivell digital i tecnològic, especialment degut a la seva influència en les nostres vides. Així, estar comunicats i amb accés a la xarxa en tot moment s'ha convertit en una necessitat vital pel desenvolupament de moltes persones, que gràcies a dispositius com *tablets* i *smartphones* s'han aconseguit suplir, però que depèn en gran mesura de la cobertura de l'operador mòbil contractat.

Concretament, aquest projecte sorgeix per tal d'oferir un ampli servei d'estadístiques i informació sobre quines són les senyals prioritàries segons l'àrea geogràfica interessada. Per realitzar-ho, s'han realitzat dues parts de desenvolupament ben diferenciades, una de les quals permet els usuaris conèixer i compartir a una base de dades les dades de senyals mòbils captades (aplicació *Android*), i la segona de les quals permet els usuaris visualitzar els resultats agrupats per regió de la intensitat de les totes senyals compartides, així com pels diferents tipus de senyals 2G, 3G, 4G (aplicació *Web*).

Dels resultats obtinguts han sorgit diferents conclusions, com ara bé que les senyals 4G tot i ser les que ofereixen una millor velocitat de connexió reben generalment una pitjor cobertura que les anteriors generacions, o com ara bé descobrir els operadors que ofereixen millor cobertura a zones "mortes" (metro, muntanya) de connexió. D'aquesta manera, s'espera que aquest servei serveixi els operadors mòbils per millorar les seves infraestructures i els usuaris per tenir coneixement de les xarxes mòbils òptimes a qualsevol regió del mapa.

Paraules clau

Escàner de xarxes d'operadors mòbils, cobertura de senyals GSM – CDMA – LTE, mapeig d'operadors mòbils, aplicació Android amb Kotlin, base de dades MySQL, i aplicació Web amb Angular2.

Abstract

Nowadays, people live in the Information Age, a time characterized by the big advances at the digital and technological levels, especially due to their influence on our lives. Thus, being connected and having constant availability to the network has become a vital necessity for the daily live of many persons, which have been able to accomplish thanks to devices such as tablets and smartphones, but that depends especially on coverage and intensity of the contracted mobile operator.

Concretely, this project arises in order to offer a broad statistical and information service about which are the priority signals according to the concerned geographic area. In order to accomplish its purpose, two well-differentiated parts of development have been carried out, one of which allows users to know and share data on the captured mobile signals (Android application), and the second one that allows the users to view the results of all Shared signals grouped by region as well as by its different type of technology: 2G, 3G, 4G (Web application).

The obtained results have derived in some conclusions, such as the fact that the 4G signals, although offering the fastest connection to the network, generally receive worse coverage than its previous generations, or as well as discovering the operators that offer better coverage under “death” areas (subway, mountain) of connection. This way, it is expectable that this service will be used by mobile operators to improve their infrastructures and also by users to get acquainted with the best mobile networks in any region of the map.

Keywords

Mobile operator network scanner, GSM – CDMA – LTE signals coverage, mobile operator mapping, Android application with Kotlin, MySQL Data Base, and Web application with Angular2.

Índex

Capítol 1: Introducció	12
1. Introducció i motivació del projecte	12
2. Descripció i tecnologies utilitzades	14
2.1 Aplicació Android.....	17
2.2 Sistema Back-End.....	15
2.3 Aplicació Front-End.....	16
3. Objectius generals	17
3.1 Objectius principals	17
3.2 Objectius secundaris.....	18
4. Metodologia i procés de treball.....	19
5. Planificació	22
5.1 Taula de fites.....	22
5.2 Diagrama de Gantt.....	22
6. Pressupost.....	25
7. Estructura de la resta del document	26
Capítol 2: Anàlisi.....	27
1. Estat de l'art	27
2. Públic objectiu	29
2.1 Perfils d'usuari	29
3. Definició d'objectius i especificacions del producte	32
Capítol 3: Disseny	33
1. Arquitectura general del sistema.....	33
2. Arquitectura de la informació i diagrames de navegació	34
2.1 Aplicació Android.....	34
2.2 Sistema Back-End.....	39
2.3 Aplicació Front-End.....	42
3. Disseny gràfic i interfícies	46
3.1 Estils	46
3.2 Usabilitat.....	47
4. Llenguatges de programació i APIs utilitzades.....	48
4.1 Aplicació Android.....	48
4.2 Sistema Back-End.....	49

4.3 Aplicació Front-End.....	50
Capítol 4: Implementació.....	51
1. Requisits d'instal·lació.....	51
2. Instruccions d'instal·lació	52
Capítol 5: Demostració	53
1. Instruccions d'ús	53
2. Tests.....	54
2.1 Aplicació Android.....	54
2.2 Sistema Back-End.....	56
2.3 Aplicació Front-End.....	57
Capítol 6: Conclusions i línies de futur.....	58
1. Conclusions	58
2. Línies de futur.....	60
Bibliografia.....	61
Annexos	62
Annex A: Glossari	62
Annex B: Lliurables del projecte	63
Annex C: Currículum Vitae	63

Figures i taules

Índex de figures

Figura 1: Percentatge d'usabilitat de telèfons intel·ligents als principals països europeus	12
Figura 2: Gràfic comparatiu de quota de mercat dels diferents sistemes operatius mòbils.....	15
Figura 3: Organització (parcial) Scrum a l'eina web Jira de les tasques a realitzar pels diferents Sprints durant el seu desenvolupament	20
Figura 4: Vista del darrer Sprint actiu durant el desenvolupament de l'aplicació Android.	20
Figura 5: Vista d'actualitzacions de codi realitzades durant el desenvolupament de l'aplicació Android	20
Figura 6: Vista d'actualitzacions de codi realitzades durant el desenvolupament de l'aplicació Back-End	21
Figura 7: Vista d'actualitzacions de codi realitzades durant el desenvolupament de l'aplicació Front-End.....	21
Figura 8: Diagrama de Gantt inicial.....	23
Figura 9: Diagrama de Gantt final	24
Figura 10: Diagrama general del sistema sencer. Android, Back-End, Front-End	33
Figura 11: Diagrama de flux de l'aplicació Android	35
Figura 12: Diagrama de navegació a partir de les interfícies de les vistes principals de l'aplicació Android.....	36
Figura 13: Exemples dels tipus de notificacions utilitzades a l'aplicació per guiar l'usuari.....	37
Figura 14: Divisió en mòduls i packages de l'aplicació Android	39
Figura 15: Part del back-office que permet revisar la informació de la base de dades mitjançant una petició HTTP GET autoritzada	41
Figura 16: Part del back-office que permet revisar el tipus de contingut que l'API REST admet per les peticions HTTP POST, i també realitzar-ne peticions autoritzades d'aquest tipus	41
Figura 17: Vistes de la pàgina web a ordinadors de 17" i 11"	45
Figura 18: Vistes de la pàgina web a mòbils iPhone 5 i Nexus 6	45

Índex de taules

Taula 1: Taula de fites relacionada amb el nombre de les PAC i la seva data d'assoliment.....	22
Taula 2: Pressupost referent a l'equip humà de desenvolupament	25
Taula 3: Pressupost referent a l'equipament tècnic necessari	25
Taula 4: Comparativa entre característiques de les diferents tecnologies de connexió mòbil	27

Capítol 1: Introducció

El propòsit d'aquest projecte és facilitar una sèrie de dades i estadístiques sobre les intensitats de les senyals rebudes pels diferents proveïdors telefònics en diverses àrees arreu el mapa. En aquesta secció es farà una introducció sobre el context del projecte, les motivacions alhora de realitzar-ho, els seus objectius i la forma de treball adoptada per portar-ho a terme.

1. Introducció i motivació del projecte

Actualment vivim a l'era de la informació, un temps caracteritzat pels grans avanços que s'han produït a nivell digital i tecnològic, especialment degut a la seva influència en les nostres vides. Així, estar comunicats i amb accés a la xarxa en tot moment s'ha convertit en una necessitat vital pel desenvolupament de moltes persones, que gràcies a dispositius com *tablets* i telèfons intel·ligents (en anglès, *smartphones*) s'han aconseguit suplir, però que depèn en gran mesura de la cobertura de l'operador mòbil contractat.

Es coneix com *smartphone* qualsevol telèfon mòbil que permet portar a terme accions pròpies d'un assistent digital personal (en anglès, PDA), més enllà de les trucades de veu i missatges SMS. El primer *smartphone* es va fabricar l'any 1992 i va ser l'IBM Simon [1]. A partir d'aquest moment es van desenvolupar noves generacions de telèfons intel·ligents, incloent gradualment noves tecnologies i funcionalitats dins dels aparells: navegador GPS, connexió WiFi, connexió per xarxa mòbil, videotrucades, reproductors multimèdia, etc. Actualment, a dates actualitzades de 12 de gener de l'any 2018, dels 7.59 billons de persones al món [2] n'hi ha 2.53 billons d'elles que utilitzen *smartphones* [3], valors que representen que el 33% (un terç) de la població mundial fa ús dels telèfons intel·ligents, independentment de la seva edat, raça o localització geogràfica. Segons enquestes realitzades a l'any 2015 [4], concretament a Espanya el valor de penetració segons la població és molt més alt que la mitja, sent el país europeu amb més usabilitat de *smartphones* amb un 69% com es pot veure a la següent gràfica:

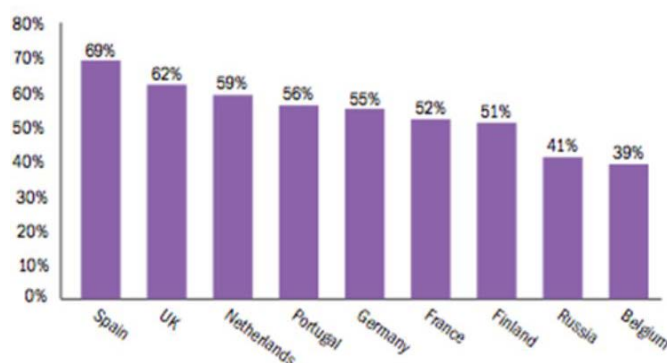


Figura 1. Percentatge d'usabilitat de telèfons intel·ligents als principals països europeus

La necessitat de comunicació i d'un accés continu a la informació va desembocar en la creació de xarxes mòbils, és a dir, xarxes de comunicació on l'últim enllaç és sense fils per dispositius mòbils. La xarxa es distribueix per àrees terrestres anomenades cel·les, cadascuna amb almenys un transceptor

de posició fixa, però normalment amb tres estacions transceptors de base, les quals proporcionen a la cel·la la cobertura de xarxa que es pot utilitzar per a la recepció i transmissió de qualsevol tipus de dada. Quan es combinen, aquestes cel·les donen cobertura de ràdio a una àmplia zona geogràfica, permetent la comunicació amb la xarxa dels dispositius intel·ligents a partir de la targeta SIM proporcionada pel seu operador mòbil [5]. Així, utilitzant aquesta tecnologia es va desenvolupar el que es coneix com internet mòbil i la importància del tràfic de dades va produir millores en les tecnologies utilitzades amb l'objectiu de millorar la velocitat i latència de transmissió de dades, generant diferents tipus de senyals que podem agrupar dins 2G, 3G i 4G.

El propòsit d'aquest projecte és agrupar informació sobre els rangs de cobertura d'aquestes senyals mòbils a una pàgina web disponible públicament en funció dels seus proveïdors basant-se en les informacions capturades d'una senzilla aplicació mòbil. Es pot dir que la seva motivació és doble: per una part, servir els usuaris per verificar les intensitats de les senyals d'operadors mòbils; i per altra part, contribuir a les companyies telefòniques per millorar les seves infraestructures i tecnologies per expandir la seva cobertura i millorar així l'experiència dels usuaris finals.

D'altra banda, a un nivell més personal, es tracta d'un projecte que servirà per tal d'ampliar coneixements en diverses plataformes i llenguatges punters en l'actualitat de la programació generant un sistema complet d'aplicació mòbil, sistema *back-end* i aplicació *front-end*. Així, tot i tractar-se d'aplicacions que es distribuïran de forma gratuïta, aquest projecte és privatiu amb la intenció de generar beneficis econòmics seguint un model de negoci basat en les relacions amb els diferents operadors mòbils en base a contractes per recomanar les seves xarxes com a prioritàries alhora de proveir-los d'un servei d'anàlisi i d'estructuració de *Big Data* segons les dades rebudes que serveixi per millorar els seus serveis d'internet mòbil.

2. Descripció i tecnologies utilitzades

La temàtica del Treball Final de Màster es tracta en recopilar la cobertura de les diferents xarxes d'operadors mòbils, que s'ha convertit en un tema d'interès a nivell d'usabilitat i informació. Les necessitats de romandre disponibles en tot moment per qualsevol aspecte comunicatiu, tant com la d'estar informats són dos problemes que s'han demostrat afectar a una gran quantitat de la població mundial, pel que és important recollir una gran quantitat de dades per tal de extreure estadístiques i conclusions que ajudin a minimitzar qualsevol aspecte que impedeixi satisfer les necessitats establertes, tant a nivell concret (usuaris amb *smartphone*) com a nivell general (operadors mòbils).

Dins d'aquest àmbit, es tracta d'un projecte que no té cap precedent que parteixi amb la mateixa consigna, tot i què sí que n'hi ha de referències similars, com ara les aplicacions *Speedtest* (que medeix la velocitat a Internet) i *Amplificador GSM* (molt similar, ja que monitoritza on són els repetidors per millorar la cobertura). No obstant, el projecte *Cobertura de Xarxes d'Operadors Mòbils (agrupades per regió)* aporta molt més que una monitorització individualitzada de les millors zones de cobertura segons el desplaçament de l'usuari, a més, genera una base de dades amb la informació extreta dels propis usuaris de l'aplicació que permet comparar les cobertures dels diferents operadors telefònics al mapa, pel que es tracta d'un projecte innovador: tot i no crear cap concepte nou, genera una solució a un problema existent. En el cas dels usuaris, millorarà la seva forma de decidir quina companyia contractar en funció de la seva cobertura; mentre que en el cas de les companyies telefòniques, millorarà la seva visió sobre les zones on han d'invertir major infraestructura per tal d'oferir serveis més òptims.

Per tal de fer-ho, s'ha portat a terme el desenvolupament d'un sistema complet basat en una aplicació mòbil *Android* i una aplicació *Front-End* (web), així com un sistema de *Back-End* que farà de nexa entre les dues. Cadascuna de les parts té una importància vital en el seu conjunt, pel que totes han de ser detallades de forma concisa.

2.1 Aplicació Android

La primera part del projecte es basa en el desenvolupament d'una aplicació Android (que actualment representa el 92% [6] de les ventes de telèfons mòbils a Espanya, com es pot observar a la Fig. 2) la qual té com a funció principal la captació de les dades de la cobertura i intensitat de l'operador de telefonia (Movistar, Orange, Vodafone...) proveïdor d'Internet mòbil del dispositiu, i mostrarà els resultats de l'escaneig per pantalla. Opcionalment, un cop obtingudes aquestes dades, es dona la possibilitat a l'usuari a compartir-les juntament amb una posició relativa d'on han set capturades utilitzant la seva ubicació GPS (en anglès, *Global Positioning System*) [7] demanant si es volen concedir els permisos pertinents de localització prèviament, de manera que aquestes dades (i cap dada privada del dispositiu o usuari) s'envien per petició POST a l'API REST amb Django que forma el bloc següent que les emmagatzemarà dins una base de dades tipus *MySQL*.

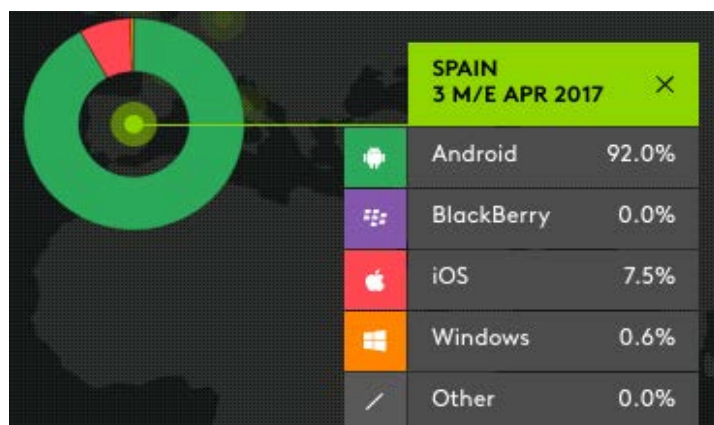


Figura 2. Gràfic comparatiu de quota de mercat dels diferents sistemes operatius mòbils

Per tal d'incentivar els usuaris a subministrar les seves dades capturades, es planteja un potencial programa de punts en futures versions de l'aplicació, que sigui directament proporcional al volum d'informació subministrada. Per exemple, rebre descomptes alhora de realitzar portabilitat a la companyia telefònica recomanada segons l'anàlisi de la base de dades en regions properes a la ubicació GPS de l'usuari, tot i què això es planteja com una opció potencial degut a la necessitat d'establiment d'acords amb consens entre les diferents companyies telefòniques.

Quant al llenguatge de programació utilitzat, s'ha pres la decisió d'utilitzar el llenguatge punter en el món Android, *Kotlin*, ja que s'ha tornat un llenguatge oficial pel desenvolupament d'aplicacions mòbils i té unes expectatives de futurs temptadores.

2.2 Sistema Back-end

Per tal de proveir la pàgina web de la informació necessària per mostrar les dades de senyal d'operadors mòbils arreu un mapa geogràfic es requereix que la informació capturada pels dispositius mòbils es comparteixi i s'emmagatzemi a una base de dades *MySQL* [8], pública per tal de inserir contingut sempre i quan sigui vàlid, i privada (mitjançant credencials d'autenticació) per tal d'obtenir totes les dades disponibles (tot i què en aquest cas, per tal de la seva visualització per part del jurat del TFM interessat s'ha mantés pública). Per tenir una major abstracció i control de la base de dades, s'ha decidit crear una API REST (de l'anglès *Representational State Transfer*), que serveix per tal de proporcionar a l'usuari una interfície entre sistemes basada en protocol HTTP, fent de pont entre client i base de dades. Així, l'API REST proporciona una arquitectura jeràrquica entre capes que separa client i servidor, independentment del llenguatge i plataformes utilitzades. En resum, l'API REST proporciona els següents mètodes per modificar les dades a la seva interfície HTTP uniforme: crear (POST), actualitzar (PUT), obtenir (GET) i eliminar (DELETE), les quals es poden configurar per efectuar-se sobre qualsevol de les taules de la base de dades a partir de l'*endpoint* establert (és a dir, la direcció o URI en anglès que identifica únicament l'acció a realitzar).

Quant a la tecnologia utilitzada per realitzar el conjunt d'API REST i base de dades s'ha decidit utilitzar *Django REST Framework*, un llenguatge basat en *Python* utilitzat precisament degut a què proporciona

de manera conjunta una base de dades SQL que rep els models de dades serialitzats i els inserta a les taules pertinents i una API REST.

2.3 Aplicació Front-end

La part final del projecte es basa en el desenvolupament d'una aplicació Front-End (web) la qual té com a funció principal la recuperació de les dades de la cobertura i intensitat de l'operador de telefonia (*Movistar, Orange, Vodafone...*) proveïdor d'Internet mòbil dels dispositius que han compartit els seus escàners, i mostrarà els resultats per pantalla. Per tal de mostrar els resultats, s'haurà de fer una petició GET amb les capçaleres necessàries per obtenir la informació emmagatzemada a l'API REST de *Django*, és a dir, s'haurà d'afegir el mecanisme d'*Intercanvi de Recursos d'Origen Creuat* (en anglès, *CORS*), per tal que l'usuari obtingui permisos per accedir a la informació. Així, la pantalla principal de l'aplicació web serà la visualització del conjunt de resultats obtinguts, que en funció dels usuaris, podran observar-se de forma gràfica a un mapa, o a una taula amb la informació complerta sobre cadascun dels escàners captats.

Quant a la tecnologia utilitzada per realitzar la pàgina web, el llenguatge de programació triat va ser *Angular2* [9]. Cal remarcar que *Angular2*, així com tot llenguatge de *JavaScript*, es complementa de HTML (de l'anglès *HyperText Markup Language*) i CSS (de l'anglès *Cascade Style Sheets*) per tal de dissenyar les aplicacions web. El llenguatge d'etiquetes HTML s'utilitza per tal de definir i elaborar el contingut d'una pàgina web (text, imatge), així com la seva organització i la seva estructura base, mentre que el llenguatge CSS és un mecanisme simple per afegir estils (fonts, colors, marges, posicions, visualitzacions) als diferents documents d'una pàgina web (és a dir, a cada etiqueta de contingut HTML).

3. Objectius generals

En els següents apartats es mostrarà un llistat i descripció dels objectius del projecte. L'ordre en què es mostraran serà: en primer lloc els objectius prioritaris, i en segon lloc els objectius secundaris que han sorgit alhora de posar en pràctica el desenvolupament del projecte.

3.1 Objectius principals

Objectius de l'aplicació Android:

- Implementar un programa capaç d'escanejar les dades de xarxa mòbil disponibles i detectar-ne canvis d'intensitat de senyal, de tipus de senyal i de proveïdor.
- Optimitzar la compilació en produccions de release, per tal que el codi no sigui *debugable*, minificar-lo i ofuscar-lo, aconseguint la mínima mida d'espai requerit i assegurant la integritat del codi.
- Arribar a la màxima audiència possible, minimitzant la versió mínima de l'aplicació a l'API 17, de manera que només un 3% dels usuaris *Android* no podran instal·lar l'aplicació (estadístiques extretes de la web oficial d'*Android developers* [10]).
- Aconseguir atreure els usuaris a que comparteixin els resultats dels seus escanejos (opcional), sempre mantenint la seva privacitat intacta.
- Assegurar el correcte funcionament de l'aplicació i prevenir errors, mitjançant testos unitaris i d'integració *end-to-end*.

Objectius de l'aplicació Web:

- Implementar una pàgina capaç d'extreure el conjunt de dades emmagatzemades a la base de dades d'escaneig de xarxes mòbils i mostrar-les per pantalla.
- Integració amb l'API de *Google Maps* per tal de mostrar els resultats amb marcadors dins el mapa del món.
- Assegurar el correcte funcionament de l'aplicació i prevenir errors, mitjançant testos manuals.

Objectius per als usuaris i companyies clients:

- Mantenir una bona experiència d'usuari durant l'aplicació Android. Oferint-li una guia d'ajuda en cas que la vulgui consultar, aconseguint guiar-lo durant tot el procés de l'aplicació i amb un disseny agradable.
- Mantenir una bona experiència d'usuari durant l'aplicació Web. Oferint les dades en un format intel·ligible i mantenint un disseny en concordança amb l'aplicació Android.
- Aconseguir oferir una informació d'utilitat tant per usuaris com per companyies proveïdores de xarxa mòbil. Que serveixi als primers per decidir la companyia a contractar o per valorar quines són les millors zones de cobertura al seu àmbit, i que serveixi als segons per tal de millorar les seves infraestructures en cas de que fos necessari.

Objectius personals de l'autor del projecte:

- Aprofundir en el coneixement dels diferents llenguatges alhora de desenvolupar aplicacions *Android*, *Back-End* i *Front-End*; investigar sobre els llenguatges punters i més òptims; i procurar realitzar un codi robust i net.
- Aplicar els coneixements de gestió de projecte adquirits al llarg del màster per tal de planificar correctament el seu desenvolupament, així com portar a terme un correcte seguiment del projecte.
- Oferir un servei d'utilitat pel qual poder sentir-se orgullós i contribuir al desenvolupament de la tecnologia la major part que sigui possible.

3.2 Objectius secundaris

- Ús de les eines web *Jira* i *Bitbucket* per la gestió del projecte.
- Ús del programari *Twine* per tal de gestionar les traduccions dels *strings* d'idioma.
- Creació de comptes de *Google Developers* i d'*Amazon Web Services* per l'obtenció d'una *API key* de *Google Maps* i per la instal·lació del sistema complet a un servidor públic.
- Automatització d'un *deploy* d'un sistema creat en local a una màquina virtual, concretament a una instància *EC2* d'*Amazon*, on s'allotjarà tant la part *back-end* com la *front-end*.
 - <http://mobile-network-scan.com>
 - <http://mobile-network-scan.com/api/networkscans>
- Compra i gestió d'un domini web, enllaçant-lo amb el servidor proporcionat per *Amazon*.
- Establir un sistema de balancejador de càrrega per evitar sobrecàrregues i caigudes del servidor.

4. Metodologia i procés de treball

La realització d'aquest projecte es basa tant en l'aplicació dels coneixements adquirits durant el transcurs del màster universitari d'aplicacions multimèdia com en l'aprenentatge autònom dels llenguatges de programació utilitzats, basada en els conceptes *CLEAN i SOLID* [11].

El primer a tenir en compte en aquest punt és l'anàlisi previ del producte que es pretén oferir. A partir d'un procés de recerca dins les aplicacions d'Android i Web, es va observar que sí que n'hi ha productes existents de molt similars, però que de cap manera s'adapten als objectius finals descrits anteriorment d'aquest projecte. D'aquesta manera, la decisió entre desenvolupar un producte nou o adaptar i ampliar un de vell va ser generar un producte totalment nou, però tenint en compte el treball previ d'anàlisi sobre les possibilitats i limitacions tècniques per portar-ho a terme. Sobre aquest punt, es va concloure que l'única dificultat era la captació de dades mitjançant un dispositiu mòbil, però mitjançant una prova de concepte es va arribar a la conclusió que era realitzable fent dos desenvolupaments paral·lels per captar la senyal: un tenint en compte que la senyal rebuda prové de l'operador contractat, i un tenint en compte que la senyal rebuda pot provenir de qualsevol operador.

Un punt realitzat l'anàlisi, es va haver de decidir una metodologia de treball i de desenvolupament, la qual va ser l'assignació de tasques en forma de desenvolupament àgil, concretament amb les bases de *Scrum*. En resum, les metodologies àgils es basen en una sèrie de principis i valors per tal d'entregar un producte final amb garanties de qualitat de manera ràpida i amb flexibilitat per realitzar-ne modificacions durant el desenvolupament. Per tal de començar a treballar, es va dividir l'objectiu final en petits desenvolupaments i implementacions, els quals conjuntament conformen cadascuna de les aplicacions, definint entregues cada aproximadament 4 setmanes. Per la seva part, el mètode *Scrum* és una implementació pràctica de la metodologia àgil.

L'aplicació de *Scrum* va suposar la creació de totes les tasques necessàries per realitzar el projecte, és a dir, la creació de la pila de producte (en anglès, *product backlog*), en funció de la seva prioritat. Totes elles dividides segons el seu entorn (*Android, Back-End, Front-End*) i establertes amb un valor representatiu de la seva dificultat, els denominats punts d'història, seguint la sèrie de Fibonacci. També es van definir els *Sprints*, és a dir, les iteracions que comporten els diferents blocs de desenvolupament i que comprenen una sèrie de tasques, de manera que cada *Sprint* d'una setmana equival a aproximadament les tasques que es puguin comprendre dins de 24-26 punts d'història. Així, es va decidir que a l'inici de cada nou *Sprint* es valorarien quines tasques són les adequades per complir-se en el següent procurant sent fidels a l'aproximació esmentada i, en cas d'haver d'incloure modificacions, es farien els ajustos necessaris a la següent iteració per tal de no alterar significativament el desenvolupament actual.

A nivell pràctic, l'eina utilitzada per complir amb els principis de la metodologia àgil i de *Scrum* establerts es tracta del *Jira*, una aplicació web que ofereix totes les eines necessàries per tal de planificar, supervisar i publicar *software*. Un exemple d'ús de l'aplicació web es pot veure a la Fig. 3, on es mostra l'estat de la pila de producte; i a la Fig. 4, on es mostra la vista àgil del darrer *Sprint* durant el seu desenvolupament.

T	Key	Summary	Assignee	P	Status	Resolution	Created	Updated	Sprint	Puntos de Historia
	MNS-1	Create app skeleton	Juan Luis Mari	↑	Resolved	Fixed	02/Oct/17	19/Oct/17	Sprint 1	5
	MNS-2	Home screen tasks relative to UI and navigation	Juan Luis Mari	↑	Resolved	Fixed	02/Oct/17	28/Oct/17		
	MNS-3	Home UI	Juan Luis Mari	↑	Resolved	Fixed	02/Oct/17	06/Oct/17	Sprint 1	2
	MNS-4	Home navigation	Juan Luis Mari	↑	Resolved	Fixed	02/Oct/17	06/Oct/17	Sprint 1	2
	MNS-5	Help screen	Juan Luis Mari	↑	Resolved	Fixed	02/Oct/17	06/Oct/17	Sprint 1	2
	MNS-6	The scan result tasks as listing results and recommending a particular option	Juan Luis Mari	↑	Resolved	Fixed	02/Oct/17	28/Oct/17		
	MNS-7	Scan result UI	Juan Luis Mari	↑	Resolved	Fixed	02/Oct/17	23/Oct/17	Sprint 3, Sprint 4	3
	MNS-8	Scan result logic	Juan Luis Mari	↑	Resolved	Fixed	02/Oct/17	23/Oct/17	Sprint 3, Sprint 4	5
	MNS-9	Instrumentation tests	Juan Luis Mari	↑	Resolved	Fixed	02/Oct/17	25/Oct/17	Sprint 4	5
	MNS-10	Network scanner implementation	Juan Luis Mari	↑	Resolved	Fixed	02/Oct/17	28/Oct/17		
	MNS-11	Network scanner Service	Juan Luis Mari	↑	Resolved	Fixed	02/Oct/17	11/Oct/17	Sprint 2	5
	MNS-12	Network scanner save data	Juan Luis Mari	↑	Resolved	Fixed	02/Oct/17	17/Oct/17	Sprint 2, Sprint 3	3
	MNS-13	Create a MemoryDataSource	Juan Luis Mari	↑	Resolved	Fixed	02/Oct/17	17/Oct/17	Sprint 2, Sprint 3	3
	MNS-14	Push notification implementation	Juan Luis Mari	↑	Resolved	Fixed	02/Oct/17	28/Oct/17		
	MNS-15	Push notification UI	Juan Luis Mari	↑	Resolved	Fixed	02/Oct/17	18/Oct/17	Sprint 2, Sprint 3	1
	MNS-16	Push notification logic	Juan Luis Mari	↑	Resolved	Fixed	02/Oct/17	19/Oct/17	Sprint 2, Sprint 3	3
	MNS-17	Configure coverage rules externally	Juan Luis Mari	↑	In Progress	Unresolved	02/Oct/17	28/Oct/17	Sprint 4, Sprint 5	5
	MNS-18	Configure strings externally	Juan Luis Mari	↑	In Progress	Unresolved	02/Oct/17	27/Oct/17	Sprint 4, Sprint 5	3
	MNS-19	Testing	Juan Luis Mari	↑	Open	Unresolved	02/Oct/17	27/Oct/17	Sprint 4, Sprint 5	13

Figura 3. Organització (parcial) Scrum a l'eina web *Jira* de les tasques a realitzar pels diferents *Sprints* durant el seu desenvolupament. Compta amb els punts d'història que indiquen la dificultat de les tasques, èpiques (les opcions sense punts) que engloben el conjunt de les següents subtasques per completar una secció i el seu estat

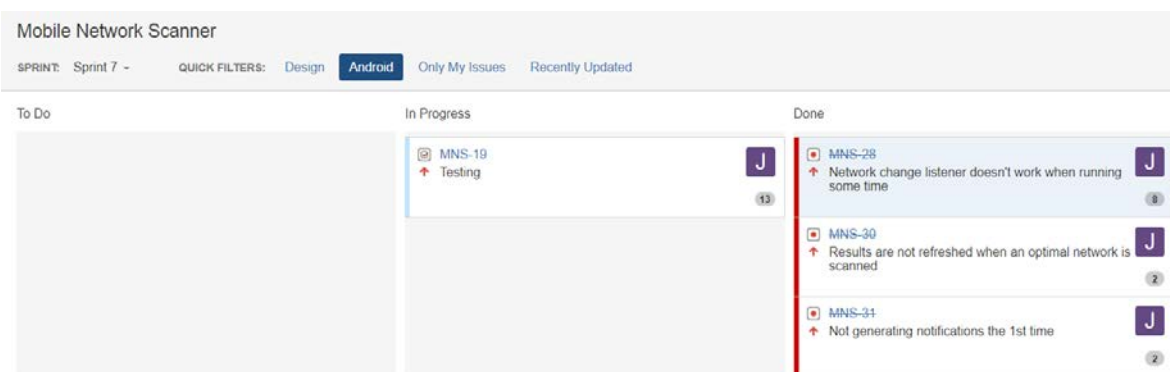


Figura 4. Vista del darrer *Sprint* actiu durant el desenvolupament de l'aplicació Android. La suma dels punts d'història és 25

També, l'eina utilitzada per tal de gestionar el desenvolupament de programari, els arxius modificats i les implementacions generades a nivell de codi és l'aplicació web *Bitbucket* que permet moltes funcionalitats com ara: la revisió del codi de manera eficient, creació de branques de desenvolupament paral·leles, distribució de versions de control, etiquetat de missatges, escalabilitat i, a més, té la possibilitat de ser enllaçat amb les tasques creades al *Jira* (veure Fig. 5 i Fig. 6).

Author	Commit	Message	Date	Builds
Juan Luis Mari	81a7a68	Merged in instrumentationTesting (pull request #13) InstrumentationTesting Approved by...	2017-10-25	
Juan Luis Mari	528cd8f	add new icons for launcher and push notifications	2017-10-25	
Juan Luis Mari	c50a86a	pr revision	2017-10-25	
Juan Luis Mari	8c43c17	add menu inside home toolbar to show the restore options	2017-10-25	
Juan Luis Mari	aea9890	add scan results activity tests	2017-10-25	
Juan Luis Mari	203881e	help tests finished	2017-10-25	
Juan Luis Mari	e25f01e	add new tests for help view	2017-10-25	
Juan Luis Mari	0742174	add more instrumentation tests	2017-10-24	
Juan Luis Mari	5c4f3bc	fix bugs	2017-10-24	
Juan Luis Mari	2da8767	add first instrumentation tests and first custom matcher	2017-10-23	
Juan Luis Mari	2ba8a7	limit the max number of items for scan results section to show	2017-10-23	
Juan Luis Mari	2a2606e	formatter and optimizer imports, also red is now light red	2017-10-23	
Juan Luis Mari	a7347d1	add _to unused variables	2017-10-23	
Juan Luis Mari	7e06a31	Merged in scanResultsLogicAndUI (pull request #12) ScanResultsLogicAndUI Approved by...	2017-10-23	
Juan Luis Mari	e944786	set weight has now default values for width and height too	2017-10-23	
Juan Luis Mari	8175531	now ToolbarBehaviour uses the toolbar context to compute statusBarHeight	2017-10-23	
Juan Luis Mari	9f41ac5	pr revision	2017-10-23	

Figura 5. Vista d'actualitzacions de codi realitzades durant el desenvolupament de l'aplicació Android

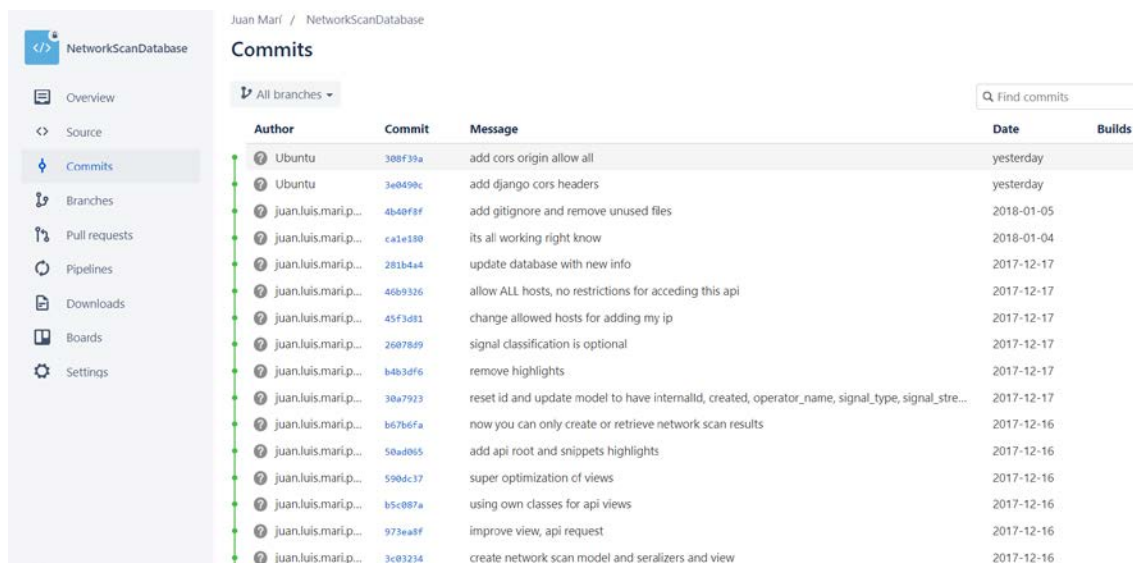


Figura 6. Vista d'actualitzacions de codi realitzades durant el desenvolupament de l'aplicació Back-End

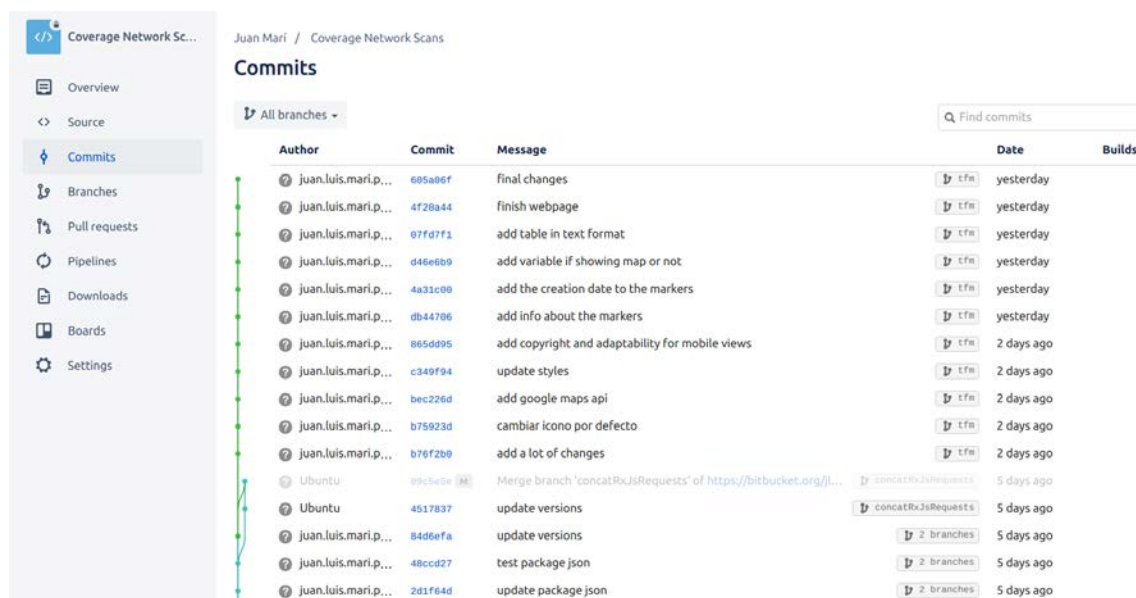


Figura 7. Vista d'actualitzacions de codi realitzades durant el desenvolupament de l'aplicació Front-End

5. Planificació

En aquest punt es tractarà la planificació de treball de forma detallada. Per fer-ho, s'indicarà a una taula de fites tots els lliuraments realitzats durant el projecte i es mostraran les tasques en detall a un diagrama de Gantt.

5.1 Taula de fites

Durant el desenvolupament del projecte s'han realitzat una sèrie d'entregues per tal de poder tenir un seguiment de la seva evolució. Aquestes es poden veure a la següent taula de fites (Taula 1).

Nom de la fita	Data assoliment
Proposta del projecte (PAC 1)	09/10/2017
Mandat del projecte (PAC 2)	28/10/2017
Lliurament parcial de l'aplicació Android (PAC 3)	25/11/2017
Lliurament parcial de l'aplicació Back-End integrada amb l'aplicació Android i base Front-End (PAC 4)	23/12/2017
Lliurament final del sistema complert. Integració final de les aplicacions (PAC 5)	14/01/2018

Taula 1. Taula de fites relacionada amb el nombre de les PAC i la seva data d'assoliment

5.2 Diagrama de Gantt

De la mateixa manera que amb la taula de fites, inicialment es va generar un diagrama de Gantt que organitzava temporalment el conjunt de tasques a realitzar per cadascuna de les parts del sistema. No obstant, durant elles es van produir una sèrie d'inconvenients (tasques noves que no s'havien plantejat per desconeixement, problemes alhora d'implementar certes parts del codi, etc.) que han derivat en lleugeres modificacions del diagrama plantejat inicialment. A les següents figures es poden veure la diferència entre els diagrames de Gantt inicial i final, entre les quals destaca la data final del desenvolupament complert (22/12/2017 → 08/01/2018, respectivament):

Diagrama de Gantt inicial:

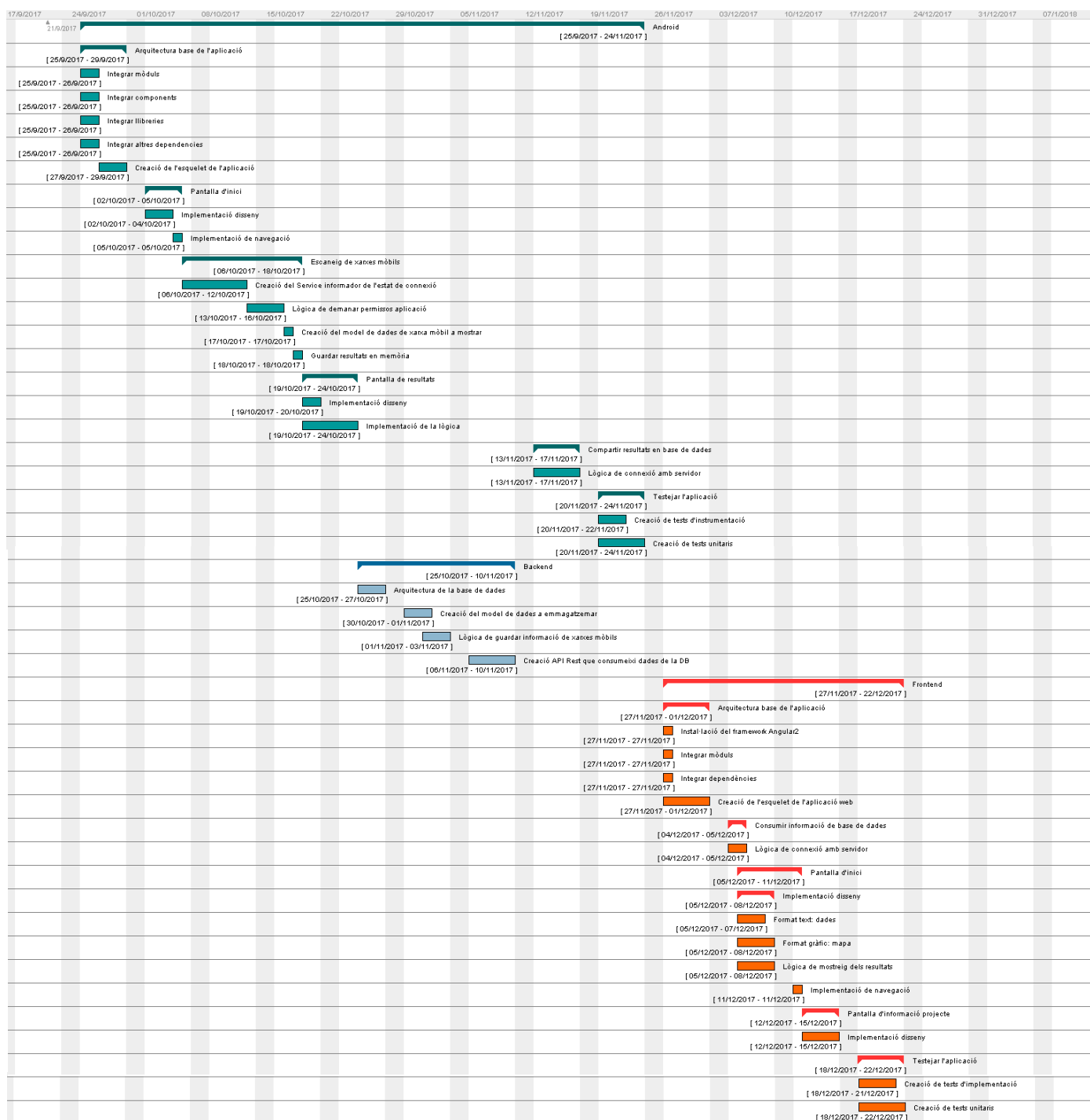


Figura 8. Diagrama de Gantt inicial

Diagrama de Gantt final:

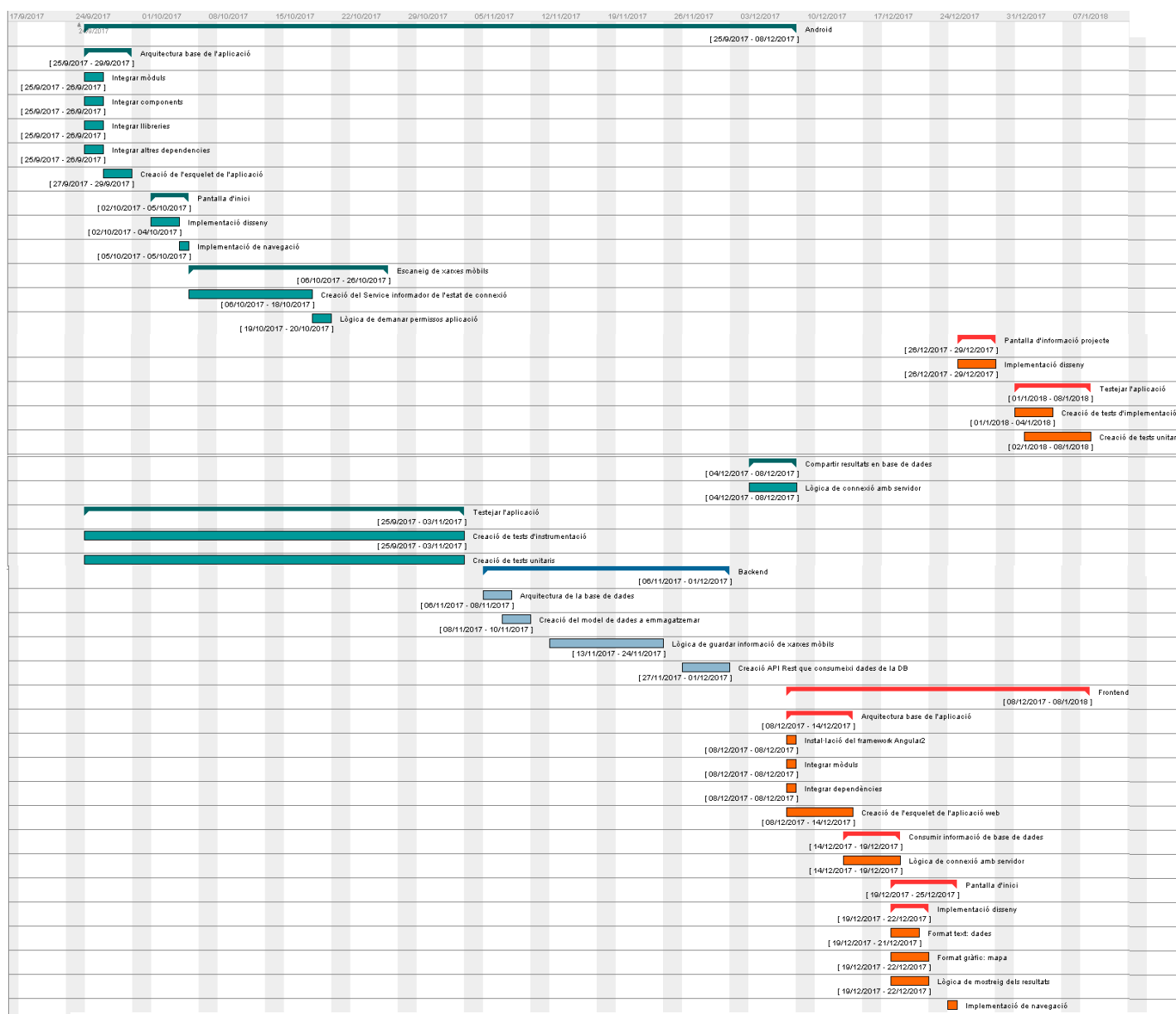


Figura 9. Diagrama de Gantt final

Cal remarcar que les modificacions més importants es varen deure a la inexperiència treballant amb sistemes tan complexes. A continuació es presenta un breu resum sobre les desviacions realitzades:

- **Primer lliurament:** 11 dies de retard. Dificultats per captar informació de xarxes i detecció de canvis a l'aplicació Android. Inexperiència amb llenguatges de *back-end* (en concret, *Django*).
- **Segon lliurament:** 4 dies de retard. Dificultats de connexió entre *front-end* i *back-end*.
- **Tercer lliurament:** 4 dies de retard. Gestions inesperades: contractació d'un servei EC2 (instància de servidor) a *Amazon*, compra d'un domini i gestions de càrrega del servidor.

6. Pressupost

El total de pressupost que ha consumit aquest projecte, fins el moment, es tracta de la suma d'equip humà de desenvolupament i equipament tècnic necessari per tal de poder desenvolupar-lo en les millors condicions possibles. Així, es poden crear dues taules en què es desglossa el cost total del projecte en funció dels principals blocs de desenvolupament i equip utilitzat:

Treball realitzat (programador Junior)	Hores	Preu / hora	Preu total
Disseny interfície gràfica	35	8	280€
Lògica d'aplicacions <i>Android, Back-End i Front-End</i>	125	20	2.500€
Integració sistema complet + <i>Amazon Web Services (AWS)</i>	50	25	1.250€
<i>Testing</i> (unitaris i d'integració)	30	10	300€
TOTAL			4.330€

Taula 2. Pressupost referent a l'equip humà de desenvolupament

Equipament requerit	Cost
Ordinador MSI GE70 2PE Apache Pro	1.090€
Compra domini <i>mobile-network-scan.com</i>	12€
Compra instància Amazon EC2	Gratis (1 any)
TOTAL	1.102€

Taula 3. Pressupost referent a l'equipament tècnic necessari

Com es pot extreure de les taules, el cost total de l'equip humà i el cost de l'equipament requerit pel correcte desenvolupament del projecte sumen $4.330 + 1.102 = 5.432€$.

7. Estructura de la resta del document

El Capítol 2 descriu la fase d'anàlisi del projecte, i inclou l'estudi de l'estat de l'art del producte a desenvolupar i un estudi de la tipologia d'usuaris que faran ús de l'aplicació.

El Capítol 3 es centra en l'etapa de disseny del projecte, explicant l'arquitectura de dades del sistema complet i els detalls d'implementació.

El Capítol 4 explica els recursos necessaris per a executar les diferents aplicacions, així com el seu procés d'instal·lació.

El Capítol 5 inclou una demostració en vídeo del funcionament de cadascuna de les parts del sistema.

El Capítol 6 recull les conclusions i les línies de futur del projecte.

Capítol 2: Anàlisi

1. Estat de l'art

Els sistemes de comunicacions mòbils van revolucionar la manera de les persones comunicar, unir comunicacions i mobilitat. Durant els darrers anys s'ha aconseguit una remarcable evolució en la història de les connexions sense fils. L'evolució de les tecnologies d'accés sense fils es troba a punt d'arribar a la seva quinta generació (5G). Mirant enrere, aquest tipus de tecnologies d'accés a la xarxa han seguit diferents camins evolutius orientat a objectius unificats: rendiment i eficiència en alta entorn mòbil. La primera generació (1G) va aconseguir l'objectiu de compartir la veu de forma bàsica, mentre que la segona generació (2G) va introduir més capacitat, cobertura i velocitat. A continuació, la tercera generació (3G) va assolir la recerca de dades a majors velocitats per obrir les portes d'una veritable experiència de "banda ampla mòbil" [12]. La quarta generació (4G) proporciona accés a una àmplia gamma dels serveis de telecomunicacions, inclosos serveis avançats de mòbils, suportats per xarxes mòbils i fixes cada vegada més velocs i amb més capacitat de transmetre paquets de bits, juntament amb un suport d'una àmplia gamma de tarifes de dades. Finalment, la planificació de la nova generació 5G té com a objectiu una capacitat superior a la 4G actual, permetent una major densitat d'usuaris de banda ampla mòbil, i donar suport a comunicacions mecàniques massives de dispositiu a dispositiu, més fiable i més gran. La recerca i el desenvolupament 5G també tenen com a objectiu una menor latència que l'equip 4G i un menor consum de bateria, per a una millor implementació d'Internet [13].

A la taula següent (Taula 4) es poden verificar els principals avanços de cada generació de xarxes mòbils:

Tecnologia	1G	2G / 2.5G	3G / 3.5G	4G	5G
Velocitat	2kbps	9.6 – 64kbps	1 – 14.4Mbps	100 – 300Mbps	(+) 1Gpbs
Tecnologia	Analògica	Digital	CDMA	WLAN	4G + WWWW
Serveis	Telefonia	Veu digital, missatgeria	Millores de qualitat d'àudio, vídeo i data	Informació dinàmica, dispositius variables	Inclusió de tecnologies d'intel·ligència artificial
Multiplexador	FDMA	TDMA / CDMA	CDMA	CDMA	CDMA
Connexió	Circuit	Circuit	Paquets	Paquets	Paquets
Xarxa	PSTN	PSTN	Paquets	Internet	Internet

Taula 4. Comparativa entre característiques de les diferents tecnologies de connexió mòbil

Un cop tenint en compte l'estat de la tecnologia en què es basa el projecte, hauria de realitzar-se un altre anàlisi sobre la competència de programari al mercat. Per fer-ho, s'ha realitzat una investigació sobre quin software similar es troba tant a la base d'aplicacions *Android* (*Google Play*) com a les diferents pàgines web, obtingudes mitjançant recerques de paraules clau similars a les del projecte:

- *Speedtest i mapes 3G, 4G i Wifi (Aplicació Android)* [14]. Aquesta és l'aplicació més semblant al projecte, ja que és una aplicació que serveix per obtenir dades de connexió Wifi i d'Internet mòbil, i té l'opció de mostrar el seguiment realitzat en forma de llistat i mapa. No obstant, la seva funcionalitat no és la mateixa, ja que en aquest cas el que mesura no són les xarxes per la seva intensitat de senyal sinó més bé per la seva latència (temps d'espera fins resposta) i per velocitat de baixada i pujada de dades. D'aquesta manera, també podem veure que, tot i ser molt semblant en quant a idea i temàtica, es diferencia en la informació que proporciona i en què només realitza un seguiment individualitzat de la connexió.
- *Informació de senyal de xarxa (Aplicació Android)* [15]. En aquest cas, es tracta d'una aplicació que ofereix la mateixa informació que s'ha proposat a aquest projecte, és a dir, nom de l'operador de xarxa, tipus de xarxa i la seva intensitat. A més, ofereix una funcionalitat extra referent a la ubicació de la torre de ràdio a la que està connectat el dispositiu. No obstant, es tracta d'una versió que no dona un valor d'intensitat subjectiu, ja que el valor en *dBm* de la intensitat de senyal s'ha de combinar amb el tipus de senyal per establir-ho correctament, ni tampoc ofereix un seguiment global de la informació ni fa cap tipus de seguiment individualitzat.
- *Testdevelocidad.es (Aplicació Web)* [16]. És el producte més similar trobat a nivell Web i simplement es tracta d'un analitzador de la velocitat de pujada i baixada de la connexió *WiFi* (no d'Internet mòbil), així com la seva latència, i també proveeix el nom de l'operador telefònic contractat, així com la direcció d'adreça IP de l'usuari.

Després de realitzar aquest anàlisi de la competència, es pot veure que sí que hi ha antecedents molt similars a l'objectiu i idea del projecte però que cap d'aquests tracta d'una manera completa el problema ni ofereix un servei més enllà de l'usuari actual que utilitza l'aplicació. A més, es pot veure la falta d'informació d'aquest tipus a aplicacions web, que centren les seves funcionalitats en el mesurament de connexions *d'Internet via router*, i no ofereixen cap tipus d'informació sobre intensitats mòbils. Així, es creu que aquest projecte pot tenir encara un *target audience* que realment estigui interessat en aquest que es troba actualment en el mercat a nivell d'explotació d'informació d'Internet mòbil ofert per les diferents companyies operadores mòbils.

2. Públic objectiu

El conjunt de públic que es pot beneficiar d'aquest projecte completament, o simplement d'alguna de les seves parts és pràcticament total. No existeix cap delimitació segons característiques demogràfiques, culturals, edat, raça, religió, economia... L'única possible limitació és el requeriment d'utilitzar, o bé un dispositiu *Android* per la captació de dades concretes d'aquell dispositiu en una zona concreta, o bé la possibilitat d'accés a la xarxa d'Internet per tal de navegar a l'aplicació web creada. No obstant, malgrat l'ampli ventall d'usuaris potencials, la realitat és que n'hi ha certs aspectes que han de compartir per tal de voler utilitzar l'aplicació en algun dels seus punts. Els més usuals serien alguns dels següents:

- Conèixer quina és la millor cobertura a diverses zones del mapa.
- Conèixer la intensitat dels diferents proveïdors telefònics en un punt i moment determinats.
- Voler millorar la cobertura a zones mortes del mapa.

Per tant, en referència als aspectes més destacats que ofereix el servei creat es poden establir una sèrie de perfils d'usuari que fan referència als usuaris que defineixen l'àmbit d'ús de l'aplicació.

2.1 Perfils d'usuari

Segons els trets característics esmentats anteriorment, es poden crear una sèrie de perfils d'usuari mitjançant una tècnica de construcció de persones fictícies. D'aquesta manera, es garanteix que el projecte no perd la referència als seus objectius però continua mantenint les necessitats de la majoria d'usuaris que es beneficiaran d'ell. Els perfils d'usuari també serveixen per ajudar a millorar la comprensió sobre el client i millorar així la qualitat del producte final.

A continuació es presenten quatre perfils d'usuari que utilitzarien aquest producte i els seus diferents escenaris, representant les diferents tipologies d'usuari que farien ús del servei d'estadístiques i dades sobre cobertura de les xarxes d'operadors mòbils agrupades per regions:

Nom	Joan Barbé	Perfil 1
Sexe	Masculí	
Edat	22	
Professió	Estudiant	
Descripció de la persona	En Joan és un jove estudiant centrat en els seus estudis d'Enginyeria Informàtica, però que vol conèixer món. Durant el seu darrer semestre a la universitat decideix fer un Erasmus i acabar el Treball Fi de Grau a Alemanya.	
Descripció de l'escenari	Degut a la seva falta d'ingressos decideix cercar una casa més aviat a les afores de Berlín, no massa ben comunicada, però que està molt bé de preu. Una de les seves preocupacions és si allà tindrà accessibilitat a Internet mòbil, necessari per comunicar-se amb la seva família i pels seus estudis. Per aquesta raó, cercant mesuradors de connectivitat troba l'aplicació web de Cobertura de Xarxes Mòbils per regió. Filtrant al mapa, veu que n'hi ha diverses senyals captades a un radi de 5km de la casa que ha trobat i totes elles d'una data de creació de fa menys d'1 mes. La connectivitat de tots els operadors de xarxa és molt baixa o nul·la, així que decideix cercar una casa a un altre lloc.	

Nom	Tania Gomes	Perfil 2
Sexe	Femení	
Edat	37	
Professió	Arquitecta	
Descripció de la persona	Tania és una treballadora d'arquitectura que porta més de 10 anys a la professió i és reconeguda pel seu esforç i la seva intel·ligència. És extravertida i amable.	
Descripció de l'escenari	Tania ha set recompensada al seu treball pel seu esforç i ha sigut promoguda a responsable de dissenys de secció. D'aquesta manera, els seus ingressos han augmentat considerablement, i per aquesta raó ha decidit mudar-se a un pis més gran amb molt més espai per les seves coses. Veu que hi ha zones mortes de connexió a la casa, així que cercant troba l'aplicació <i>Android</i> que permet mesurar les intensitats de tots els proveïdors telefònics disponibles, tot i no tenir-los contractats. Analitzant a diverses zones de la seva casa, descobreix que Orange és qui millor cobertura té a totes les zones, molt millor que la del seu contracte amb Vodafone, arribant sempre a una intensitat de senyal d'entre "mig" i "alt", de manera que decideix canviar de companyia.	

Nom	Damià Obrador	Perfil 3
Sexe	Masculí	
Edat	33	
Professió	Tècnic de Movistar	
Descripció de la persona	En Damià és un treballador de Movistar, que es dedica a realitzar instal·lacions i millorar les infraestructures de la companyia a Barcelona, però que no té suficients diners per remodelar el seu pis.	
Descripció de l'escenari	Després de moltes queixes per les constants pèrdues de clients de Movistar en benefici d'altres companyies emergents (Jazztel, Yoigo, Peponet, etc.), des de la junta tècnica de Movistar decideixen prendre mesures per tornar a millorar les seves prestacions i guanyar la clientela perduda. A la taula es proposa utilitzar un contracte temporal en relació amb aquest projecte, que recomanarà als usuaris les zones on Damià està treballant per millorar la infraestructura i que ell mateix ha analitzat segons l'aplicació web que comprenien zones mortes. Mesos després Movistar escala diverses posicions i incrementa els seus beneficis en un 20%, pel que Damià rep la recompensa desitjada per tal de realitzar la remodelació del seu pis.	

Nom	Rosana Pardo	Perfil 4
Sexe	Femení	
Edat	57	
Professió	Agricultora	
Descripció de la persona	Rosana és una treballadora de camp des de ben petita, que no està d'acord amb el creixement tecnològic actual i es mostra bastant descontenta amb la influència de les tecnologies a la seva vida. Sempre ha pensat que les ones són perjudicials per a la seva salut.	
Descripció de l'escenari	Rosana decideix que és hora de cercar una casa, amb proximitat amb el camp, on tenir animals que tinguin espai per jugar i per divertir-se, i alhora plantar arbres i viure de matèria prima. Així, el seu fill Jesús li cerca un lloc on no hi hagi cap antena aprop ni arribi cap senyal més enllà de les 1G - 2G (requerides per trucades telefòniques). Finalment troba un lloc a un poblat a prop de Sant Gervasi, gràcies a l'aplicació web de Cobertura de Xarxes Telefòniques i la seva capacitat per tenir un filtre segons els modes de xarxa (2G, 3G, 4G) i gràcies, paradoxalment, a la tecnologia es separa d'ella.	

3. Definició d'objectius i especificacions del producte

L'objectiu del projecte es tracta d'un servei amb l'únic incentiu d'oferir la possibilitat als seus usuaris potencials de compartir de forma anònima les dades telefòniques de la xarxa mòbil que estan consumint per tal de generar estadístiques genèriques de cobertura de les diferents tipus d'operadores mòbils emeses a qualsevol situació geogràfica.

Per tant, es tracta d'aconseguir oferir la informació necessària sobre les xarxes d'operadors mòbils disponibles al *target audience* a qui s'orienta aquesta aplicació, que segons s'ha esmentat en el punt anterior es tracta d'aquelles persones que estan cercant informació per canviar la seva tarifa telefònica, que volen establir una nova (p.e. mudança a un poble nou) o que simplement volen identificar en quins llocs poden tenir una comunicació fructífera i a quins no. O per altra banda, aconseguir contractes i acords amb *partners* interessats amb la idea i amb companyies telefòniques, de forma que sorgeixin acords per tal de beneficiar ambdues parts (el que es denomina com contractes *win-win*), rebent una part econòmica a canvi d'ajuda per recomanacions/captacions d'usuaris a la companyia operadora.

En versions futures s'espera ampliar l'idioma al català i el francès, ja que l'aplicació és multi-llenguatge (actualment castellà i anglès) i pretén assolir tot el mapa amb les seves recopilacions de dades.

Capítol 3: Disseny

1. Arquitectura general del sistema

El sistema complet es pot dividir en tres parts: l'aplicació Android, el sistema Back-End i l'aplicació Front-End. A nivell general, l'arquitectura és pot dividir en dues parts ben diferenciades:

- En primer lloc, l'aplicació Android s'encarrega de captar dades de xarxes mòbils suficients per crear un model de dades que s'ha anomenat durant tot el procés "NetworkScan", és a dir, resultats d'escaneig de xarxes mòbils: data de creació, nom de l'operador, tipus de senyal, intensitat de la senyal i les dades d'ubicació on s'ha pres l'escaneig. Aquesta transmet el conjunt de *NetworkScans* presos a l'API REST de *Django* mitjançant una petició HTTP POST, i és aquí el moment en que s'emmagatzema la informació a la base de dades *MySQL*.
- En segon lloc, l'aplicació Web s'encarrega de fer una petició HTTP GET per demanar el conjunt d'informació acumulada a la base de dades, de manera que rep el conjunt de tots els resultats d'escaneig ordenats per la seva data de creació. En funció del mode seleccionat per l'usuari de la pàgina web, aquests es mostraran en un format o un altre, mapa o text.

En resum, podem veure que el procés a nivell conceptual és simple: L'aplicació Android envia informació d'escaneig al servidor per tal que l'aplicació Front-End les mostri més endavant. A la següent figura (Fig. 8) es mostra un diagrama general del sistema.

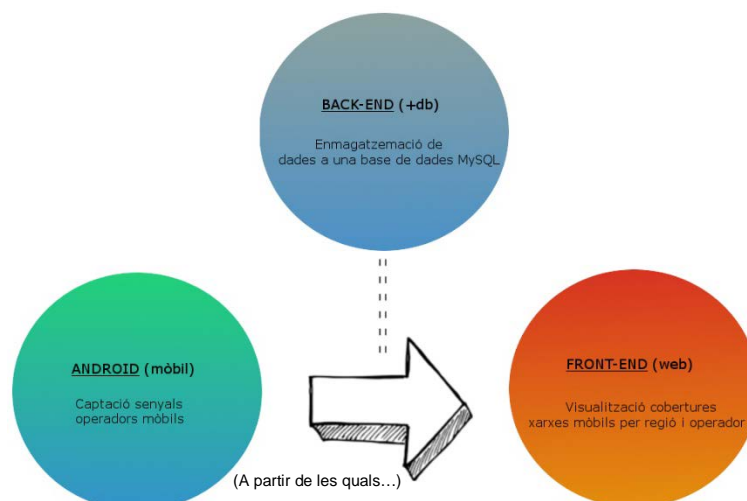


Figura 10. Diagrama general del sistema sencer: Android, Back-End, Front-End

2. Arquitectura de la informació i diagrames de navegació

En aquesta secció s'inclou la descripció detallada mitjançant diagrames dels elements principals que componen l'aplicació.

2.1 Aplicació Android

L'aplicació Android desenvolupada amb *Kotlin* (el nou llenguatge oficial de Google per desenvolupament mòbil) té com a objectiu principal la captació de la intensitat de xarxa mòbil en ús. A més, si l'usuari no només vol saber la intensitat de la seva companyia contractada, l'aplicació permet realitzar un escaneig exhaustiu del conjunt de xarxes disponibles per tal de realitzar una comparació general sobre quines són les xarxes i tipus de senyals que reben una millor cobertura a la seva localització. L'usuari, de forma voluntària, pot arribar a compartir les dades del seu escaneig a una base de dades SQL, a la qual s'accedirà mitjançant l'*API REST Django*), per tal d'aportar unes estadístiques generals a nivell geogràfic sobre les intensitats dels diferents operadors de xarxa mòbil.

Llavors, es tracta d'una aplicació amb una interfície que s'ha cercat que sigui senzilla, i la dificultat de la qual predomina en les tasques que formen la lògica per tal d'obtenir les dades de la xarxa mòbil necessàries. Un aspecte molt important relacionat amb aquest tema és que el sistema operatiu d'*Android* no permet i, per tant, no aporta cap mètode per tal d'obtenir informació de totes les xarxes mòbils disponibles, sinó que només proporciona les dades de la xarxa a la que s'està connectat en l'instant que es verifica mitjançant una instància del seu objecte *TelephonyManager*. Per tant, si l'usuari vol informació sobre un altre operador mòbil o sobre un altre tipus de senyal, l'aplicació li proporcionarà un enllaç amb la vista d'ajustos de xarxa mòbil on podrà canviar a la xarxa mòbil convenient i, mitjançant un servei en segon pla, capturarà la informació de la nova xarxa en quan es detecti un canvi de senyal.

D'aquesta manera, les vistes de les que disposarà l'aplicació seran només tres, per tal de facilitar la interacció de l'usuari amb el propòsit de l'aplicació:

1. La vista principal. És la primera pantalla que es veu cada cop quan s'inicia l'aplicació, on es mostra l'ordre preferent de xarxes mòbils a escanejar (configurable externament, la intenció és que segons els convenis amb els diferents operadors mòbils aquests valors preferents es puguin canviar dinàmicament) i els enllaços a la vista d'ajuda i a l'aplicació interna del sistema per escanejar xarxes mòbils.
2. La vista d'ajuda. És la pantalla que informa sobre el funcionament de l'aplicació per tal que l'usuari entengui quines són les pautes a seguir (ja que degut a les dificultats per captar la intensitat de senyals no connectades pot ser ambigu com obtenir-la) per tal de mesurar la intensitat de senyal de les xarxes desitjades.
3. La vista de resultats. És la pantalla que finalitza el procés, i a la qual només es pot arribar si l'usuari ha *clickat* al botó de començar l'escaneig de la vista principal i ha obtingut resultats. Un cop retorni a l'aplicació, aquesta redirigirà a la vista que mostrarà els resultats de l'escaneig

realitzat, els quals estaran classificats en tres categories (*recomanat*, *regular*, *no recomanat*) segons el rang de decibels en què es troben, partint de la següent divisió:

- Rang (0, -85] dBm = Recomanat
- Rang (-85, -97) dBm = Regular
- Rang [-97, -108) dBm (o fora de rang) = No recomanat

El diagrama de flux corresponent al conjunt de totes les accions a realitzar dins de l'aplicació és el següent:

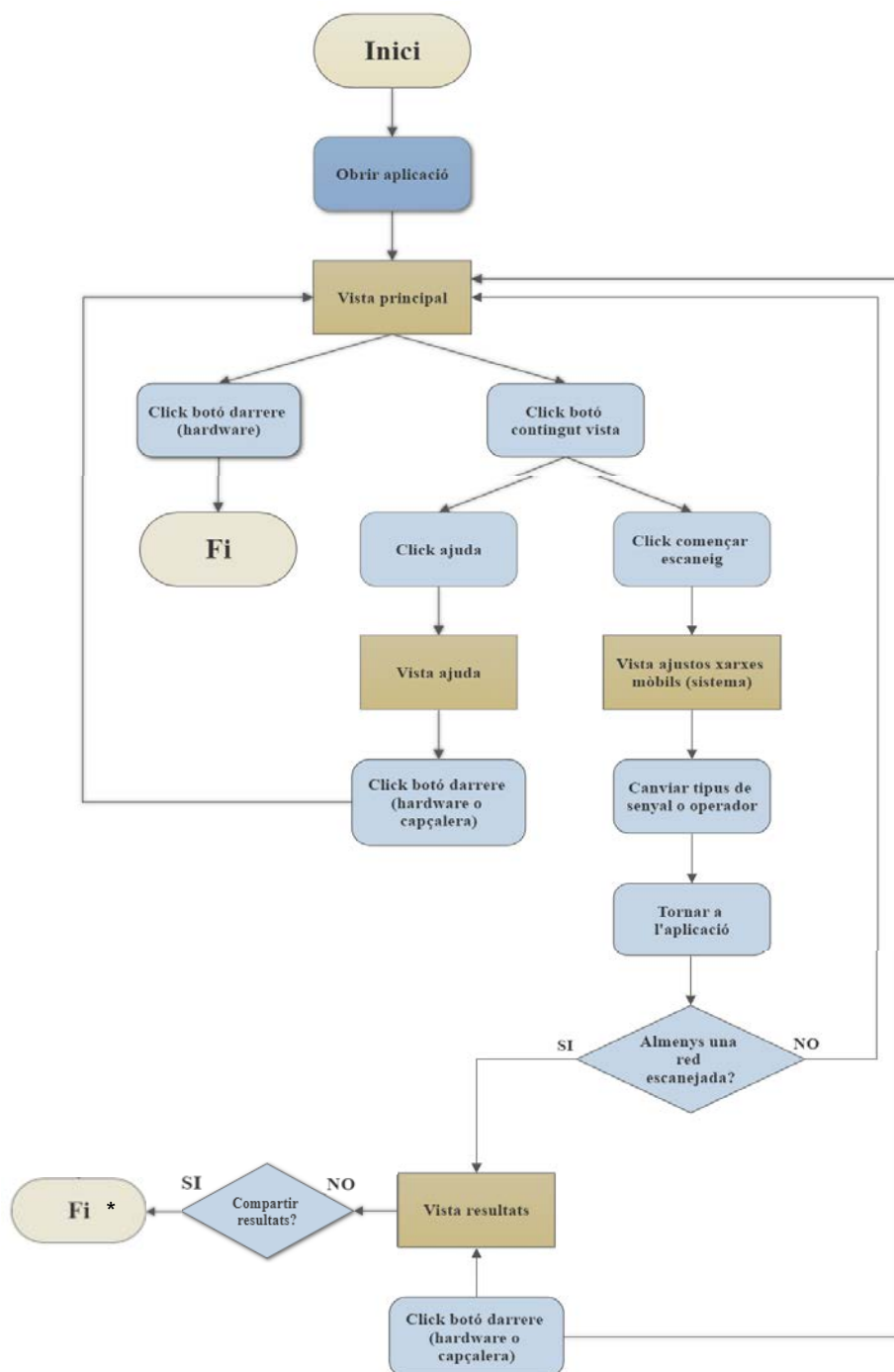


Figura 11. Diagrama de flux de l'aplicació Android

(*) Un cop es comparteixen resultats no es tanca l'aplicació, però finalitza el flux desitjat complet.

Aquest flux de pantalles segons les decisions preses i els resultats obtinguts també es pot representar segons el diagrama de navegació (mostrant les interfícies realitzades) de les vistes esmentades següent:

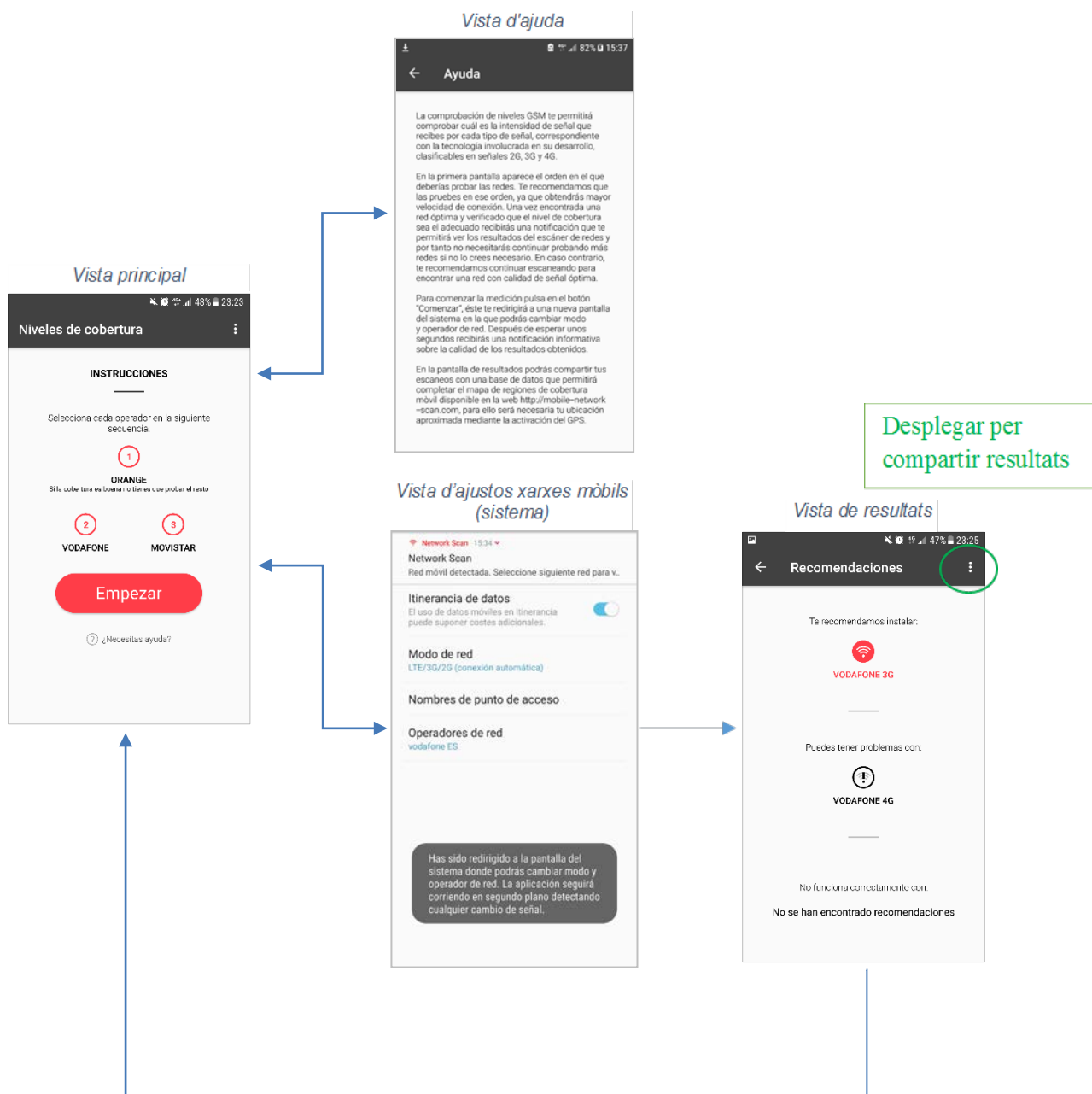


Figura 12. Diagrama de navegació a partir de les interfícies de les vistes principals de l'aplicació Android

Com es pot veure representat en qualsevol dels dos diagrames, tot comença a la pantalla de suggeriment de xarxes mòbils i, si el procés s'ha completat de la forma desitjada, acabaria un cop s'hagin compartit els resultats amb la base de dades. Tal com està plantejat el projecte, les preferències són configurables externament, i per tant, la idea és que aquestes preferències siguin variables depenent de les relacions i acords amb les diferents companyies que vulguin participar amb el projecte, de manera que a cada zona geogràfica es proposarien unes preferències diferents segons quina companyia hagi mostrat més interès en captar usuaris de la zona.

També cal esmentar un aspecte afegit, i és que aquesta navegació tracta de corregir a nivell d'usuari l'impediment de poder escanejar totes les xarxes mòbils disponibles alhora. Per fer-ho, a la pantalla d'ajuda s'informa a l'usuari sobre un resum breu del funcionament de l'aplicació:

[help_main_text]

es = La comprobación de niveles GSM te permitirá comprobar cuál es la intensidad de señal que recibes por cada tipo de señal, correspondiente con la tecnología involucrada en su desarrollo, clasificables en señales 2G, 3G y 4G.

En la primera pantalla aparece el orden en el que deberías probar las redes. Te recomendamos que las pruebes en ese orden, ya que obtendrás mayor velocidad de conexión. Una vez encontrada una red óptima y verificado que el nivel de cobertura sea el adecuado recibirás una notificación que te permitirá ver los resultados del escáner de redes y por tanto no necesitarás continuar probando más redes si no lo crees necesario. En caso contrario, te recomendamos continuar escaneando para encontrar una red con calidad de señal óptima.

Para comenzar la medición pulsa en el botón "Comenzar", éste te redirigirá a una nueva pantalla del sistema en la que podrás cambiar modo y operador de red. Después de esperar unos segundos recibirás una notificación informativa sobre la calidad de los resultados obtenidos.

En la pantalla de resultados podrás compartir tus escaneos con una base de datos que permitirá completar el mapa de regiones de cobertura móvil disponible en la web <http://mobile-network-scan.com>, para ello será necesaria tu ubicación aproximada mediante la activación del GPS.

en = The GSM level test will allow you to check the signal strength that you receive for each type of signal, corresponding to the technology involved in its development, classifiable in 2G, 3G and 4G signals.

On the first screen appears the order in which you should test the networks. We recommend that you try them in that order, since you will get faster connection speed. Once you have found an optimal network and verified that the level of coverage is adequate you will receive a notification that will allow you to see the results of the network scanner and therefore you will not need to continue testing more networks if you do not believe it necessary. Otherwise, we recommend you continue scanning to find a network with optimal signal quality.

To start the scan, click on the "Start" button, this will redirect you to a new screen of the system where you can change the network mode and operator. After waiting a few seconds you will receive an informative notification about the quality of the results obtained.

At the results screen you could share your scanners with a data base used to complete the region-grouped coverage signal map available in the website <http://mobile-network-scan.com>, for that, it will be needed your approximate location through the activation of the GPS.

De la mateixa manera, quan l'usuari va seguint el flux esmentat, diverses notificacions (*missatges toast, alertes, diàlegs informatius, notificacions push*) van apareixent dinàmicament per informar-lo de la situació en què es troba i guiar-lo correctament. Alguns exemples es poden trobar a la Fig. 13.

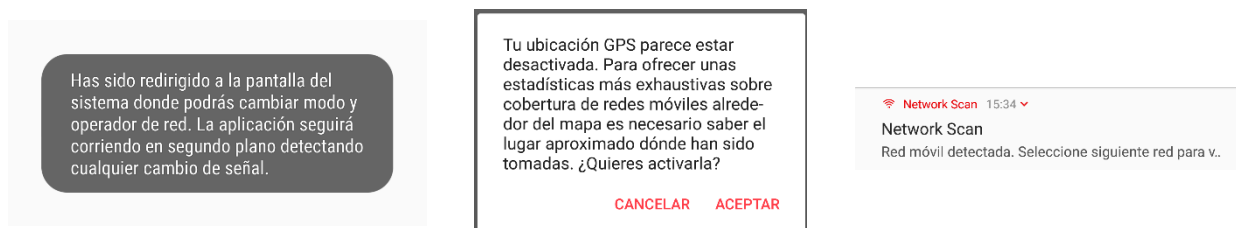


Figura 13. Exemples dels tipus de notificacions utilitzades a l'aplicació per guiar l'usuari

Cal esmentar que les dades compartides en cap moment es guarden una relació amb el dispositiu o cap informació de l'usuari del dispositiu (de fet, en cap moment es demana cap dada seva), això s'ha realitzat per tal de respectar la intimitat i anonimat dels usuaris que decideixin contribuir amb la

publicació de les seves dades estrictament relacionades amb la senyal de xarxa mòbil. El model de dades que s'obté de realitzar la cerca de xarxes mòbils disponibles és el següent:

```
data class NetworkOperator(val id: String,
                           val name: String,
                           val signalType: String,
                           val signalStrength: Int,
                           val signalClassification: SignalStrengthClassification) {

    enum class SignalConnection constructor(val value: String) {
        TYPE_2G("2G"),
        TYPE_3G("3G"),
        TYPE_4G("4G")
    }

    enum class SignalStrengthClassification constructor(val boundary: Int) {
        NONE(-108),
        VERY_LOW(-97),
        LOW(-90),
        MEDIUM(-85),
        HIGH(0)
    }
}
```

Aquest model, conjuntament amb les dades de latitud i longitud, genera un altre model: *NetworkScan*, que és el conjunt de dades que s'enviarà mitjançant la petició HTTP GET a l'endpoint de l'API REST:

```
data class NetworkScan(val id: String,
                       val name: String,
                       val signalType: String,
                       val signalStrength: Int,
                       val signalClassification: SignalStrengthClassification,
                       val latitude: Double,
                       val longitude: Double);
```

Finalment, pel que l'estructura de l'aplicació, aquesta està dividida en tres mòduls (veure Fig. 14):

- **“app”**: Al mòdul d'aplicació es troben les vistes i tota la lògica associada a elles, com per exemple: arxius de traduccions, icones, colors, navegació entre vistes, etc. Dintre seu, els diferents *packages*, es divideixen segons:
 - o *Generals (application, base i common)*
 - o *Pantalles (home, help i scanResults)*
 - o *Gestors (network i wifi)*
- **“core”**: Al mòdul anomenat nucli es troba la lògica de presentació de dades per pantalla i interaccions amb les diferents fonts d'informació (en el nostre cas, memòria i l'API REST de Django). Dintre seu, els diferents *packages*, es divideixen segons:
 - o *Generals (application, base i common)*
 - o *Pantalles (home, help i scanResults)*
 - o *Gestors (network i wifi)*
- **“domain”**: Al mòdul de domini es troben els models de dades i la informació necessària per connexions amb les diferents fonts d'informació, com per exemple les seves *URLs* o *endpoints*. Dintre seu, els diferents *packages* es divideixen segons la seva classificació (fonts d'informació, interceptors de respostes de l'API a la que s'apunta, models de dades i repositori), així com un paquet de funcions comunes.

Tots tres mòduls tenen una sèrie de tests (*androidTest*, per pantalla, i *test* per testos unitaris).

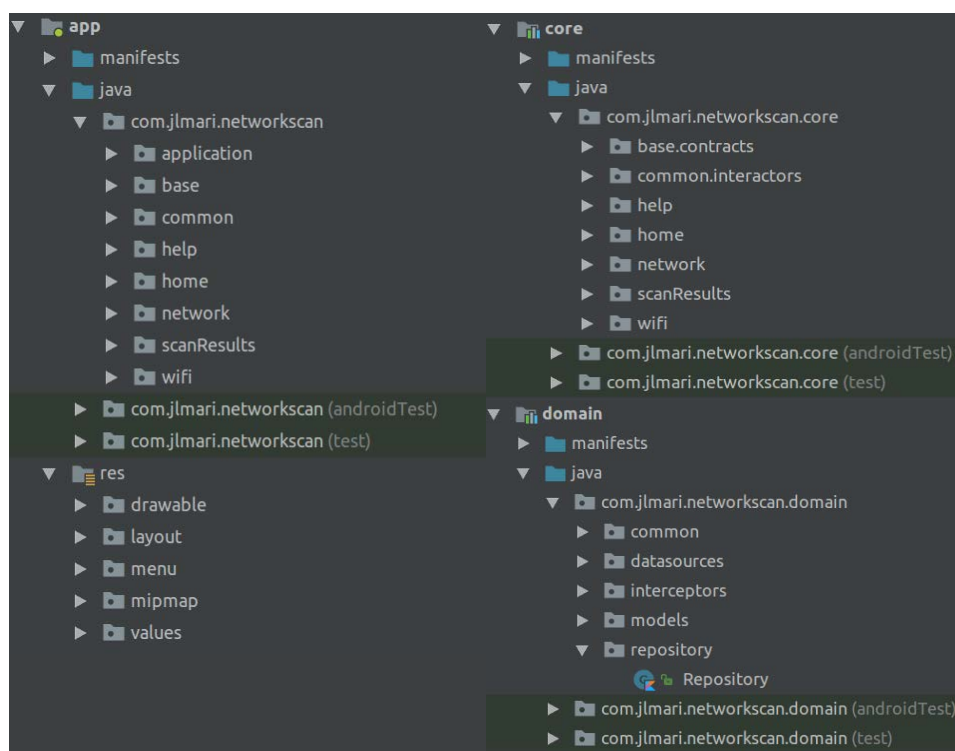


Figura 14. Divisió en mòduls i packages de l'aplicació Android

2.2 Sistema Back-End

Django posa especialment èmfasi en una estructura bàsica, simple i organitzada de tres components: els *routers* (eina per definir les direccions URI de l'API generada d'una forma simple, així com els mètodes - generar, esborrar, obtenir - o accions a realitzar per cadascuna d'elles), les vistes/*views* (que gestionen el flux des de que reben les peticions o *request* del client fins que retornen les respostes o *response* del servidor) i els serialitzadors/*serializers* (que s'encarreguen de transformar les dades de la petició i processar-les de la manera apropiada segons el contingut i el tipus de petició). Cal concretar que a aquest projecte es farà un ús molt senzill de Django, ja que per una part només hi haurà:

- Dues especificacions de *routers*, ja que només existirà una direcció *d'endpoint*, que serà la *URL del servidor + /api/networkscans/*, i que servirà tant per fer el GET (que retorna la llista sencera d'escàners: *NetworkScanList*) com per fer el POST d'un llistat de resultats d'escaneig de xarxes mòbil en detall (*NetworkScanDetail*).

```
urlpatterns = [
    url(r'^api/networkscans/$', views.NetworkScanList.as_view()),
    url(r'^api/networkscans/(?P<pk>[0-9]+)/$',
        views.NetworkScanDetail.as_view()),
]

urlpatterns = format_suffix_patterns(urlpatterns)
```

- Dues vistes: una per generar i una per retornar el contingut de resultats de l'escaneig. Aquestes són les que defineixen els mètodes *HTTP* permesos per cada model de dades. En el cas del model *NetworkScan* creat, es pot veure que els mètodes són de creació de la llista (*ListCreateAPIView: GET*) i de retribució d'un conjunt de models en detall (*RetrieveAPIView: POST*):

```
class NetworkScanList(generics.ListCreateAPIView):
    queryset = NetworkScan.objects.all()
    serializer_class = NetworkScanSerializer
```

```
class NetworkScanDetail(generics.RetrieveAPIView):
    queryset = NetworkScan.objects.all()
    serializer_class = NetworkScanSerializer
```

- Un serialitzador adjunt a un model de dades per transformar les dades de les peticions en columnes de la taula de base de dades on s'emmagatzemin el llistat de tots els resultats obtinguts o de forma inversa.

Finalment, només falta definir com funciona aquest serialitzador i quin és el model de dades definit i acceptat per la nostra API, és a dir, quin és el format i els camps que retorna quan es tracta d'obtenir dades d'ella i quin és el format i camps que accepta quan es tracta d'inserir nous resultats dins d'ella.

*) Per una part el model actual té una sèrie de camps, cadascun d'un tipus i una longitud fixats i tots ells obligatoris (menys el seu identificador i data de creació, que són auto-generats per la pròpia API):

operator_name (text de màxim 30 caràcters: Vodafone, Movistar, etc.)

signal_type (text de màxim 2 caràcters: 2G/3G/4G)

signal_strength (valor numèric llarg, pot i ha de ser negatiu: -72, -87, -102 dBm)

signal_classification (text de classificació de la senyal, màxim 20 caràcters: HIGH, LOW, etc.)

latitude (valor numèric llarg, pot ser negatiu i positiu: 40.099283, -34.19273)

longitude (valor numèric llarg, pot ser negatiu i positiu: 2.732998, -6.12398)

```
class NetworkScan(models.Model):
    created = models.DateTimeField(auto_now_add=True)
    operator_name = models.CharField(max_length=30, blank=False)
    signal_type = models.CharField(max_length=2, blank=False)
    signal_strength = models.FloatField(blank=False)
    signal_classification = models.CharField(max_length=20, blank=True)
    latitude = models.FloatField(blank=True, default=0)
    longitude = models.FloatField(blank=True, default=0)

    class Meta:
        ordering = ('created',)
```


*) Per altra part, el serialitzador s'ha creat amb la intenció de que tots els camps d'escaneig que es rebin mantinguin el mateix nom a la base de dades, és a dir, s'encarrega de l'equivalència de valors, segons el seu ordre i valor.

```
class NetworkScanSerializer(serializers.ModelSerializer):
    class Meta:
        model = NetworkScan
        fields = ('id', 'created', 'operator_name', 'signal_type',
                'signal_strength',
                'signal_classification', 'latitude', 'longitude')
```

A les següents figures es observar el disseny de la pàgina del *back-office* (o “porta del darrere”), per la gestió de tota la part relacionada amb l'API REST i la base de dades. No hauria de ser pública, però en aquest moment s'ha decidit així per tal que tothom pugui comprovar el seu funcionament correcte.

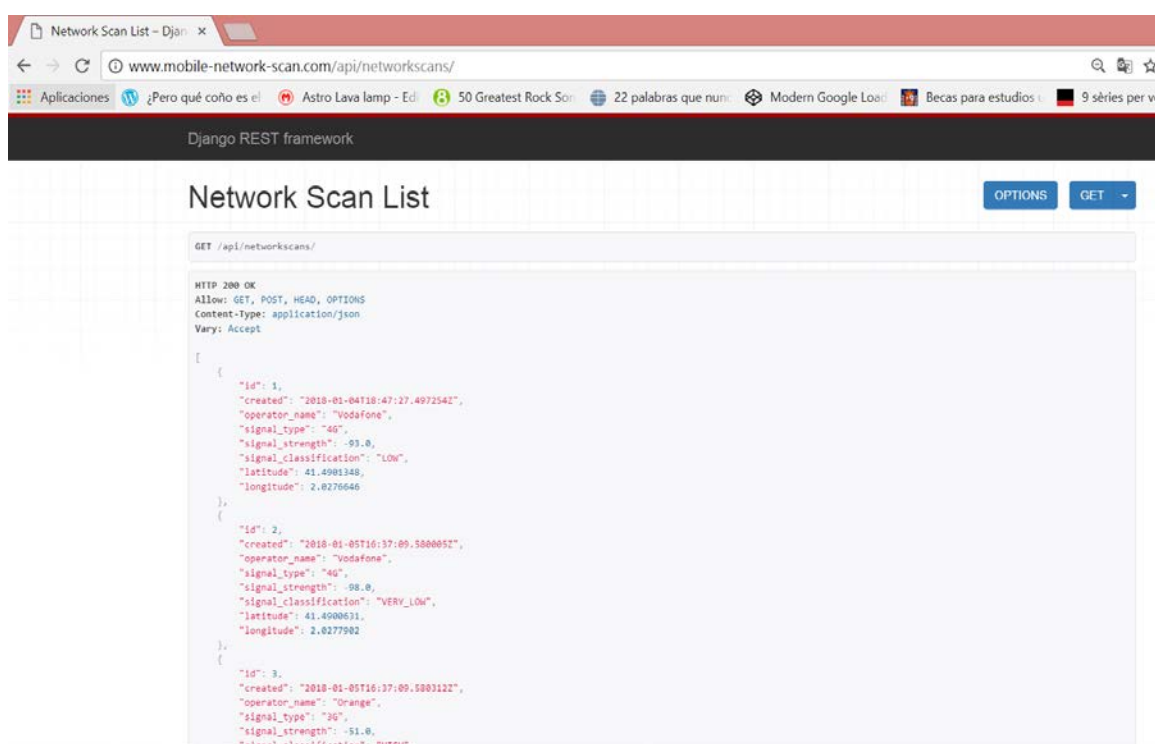


Figura 15. Part del back-office que permet revisar la informació de la base de dades mitjançant una petició HTTP GET autoritzada

The screenshot shows a web browser window displaying the Django REST framework interface for the 'Network Scan List' endpoint. The URL is 'www.mobile-network-scan.com/api/networkscans/'. The interface shows the HTTP 200 OK response with the following details:

```
HTTP 200 OK
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept
```

The response body is a JSON array of three network scan objects:

```
[
  {
    "id": 1,
    "created": "2018-01-04T18:47:27.497254Z",
    "operator_name": "Vodafone",
    "signal_type": "4G",
    "signal_strength": -93.0,
    "signal_classification": "LOW",
    "latitude": 41.4981348,
    "longitude": 2.8276646
  },
  {
    "id": 2,
    "created": "2018-01-05T16:37:09.580652Z",
    "operator_name": "Vodafone",
    "signal_type": "4G",
    "signal_strength": -98.0,
    "signal_classification": "VERY_LOW",
    "latitude": 41.4900631,
    "longitude": 2.8277982
  },
  {
    "id": 3,
    "created": "2018-01-05T16:37:09.580312Z",
    "operator_name": "Orange",
    "signal_type": "3G",
    "signal_strength": -51.0,
    "signal_classification": "HIGH"
  }
]
```

Figura 16. Part del back-office que permet revisar el tipus de contingut que l'API REST admet per les peticions HTTP POST, i també realitzar-ne peticions autoritzades d'aquest tipus

2.3 Aplicació Front-End

Angular2 és el llenguatge triat per desenvolupar la pàgina web on es mostraran els resultats finals de tot el flux d'informació des del moment que es captura en els diferents dispositius dels usuaris que decideixen compartir els resultats de les xarxes mòbils, passant per l'API fins l'emmagatzematge a la base de dades, i fins que s'arriben a *printar* dins dels marcadors geo-localitzats dins un mapa obtingut a través de l'API de Google Maps. D'aquesta manera, es pretén millorar l'experiència dels usuaris en relació amb la ubicació dels senyals mòbils rebuts i tota la seva informació intrínseca.

Sobre el desenvolupament d'aquesta aplicació, les parts més destacades són les següents:

- a) Creació del mòdul base de l'aplicació. És la part del projecte on es defineixen el conjunt de declaracions de components, proveïdors i dependències que s'utilitzaran durant el seu desenvolupament i que, per tant, són comuns a tots els components en què deriva l'aplicació. Podem veure que per aquesta aplicació s'utilitzen els següents:

```
@NgModule({
  declarations: [
    AppComponent,
  ],
  imports: [
    BrowserModule,
    FormsModule,
    HttpClientModule,
    MaterialModule,
    FlexLayoutModule,
    BrowserAnimationsModule,
    AgmCoreModule.forRoot({
      apiKey: 'AIzaSyATERDXfGDyx1h5Du7UBMrTFyxNDAZR1pA'
    })
  ],
})
```

Sobre les declaracions cal destacar que l'aplicació només es compon d'una única vista, la pàgina principal. És per això que només disposarà d'un únic component, relacionat amb un servei per demanar la informació de cobertura del conjunt de xarxes mòbils a l'API REST i que s'encarregarà de la renderització dels resultats per pantalla, així com la lògica associada a la mateixa pantalla. Dels imports es pot destacar l'ús d'un mòdul HTTP per les crides *Ajax* a la base de dades, l'ús de "*Material Design*" pel disseny gràfic de l'aplicació, l'ús d'animacions i elements flexibles pel desenvolupament responsiu (*responsive design* en anglès) de la pàgina i, finalment, l'ús de *AgmCoreModule* (*AGM = Angular Google Maps*) que es tracta d'una implementació pròpia d'Angular per tal de proveir Google Maps de forma senzilla al codi HTML mitjançant la seva directiva *agm-maps*. En aquest cas, tindrà tants *markers* com conjunt de *networkscans*, això es podrà veure a la secció HTML de l'apartat b).

- b) Creació del component base de l'aplicació. Com s'ha esmentat en el punt anterior, el component principal de l'aplicació serà l'encarregat de gestionar els aspectes de la vista (HTML) que es mostren, en funció de les dades rebudes pel servidor i les accions d'usuari que es produeixen. Per tant, podem dir que aquest component està relacionat amb:
 - a. L'arxiu HTML que defineix la distribució del contingut de la pàgina web:

i. Capçalera o Toolbar

iii. Footer

ii. Contingut base: Mapa o taula

iv. Opcionalment: Diàleg modal

```

<md-toolbar color="primary"> Cobertura de redes móviles ... </md-toolbar>
<md-sidenav-container fxFlex>
  <div class="content" fxLayoutGap="16px">
    <input type="button" value="Mapa" (click)="showMapView = true" />
    <input type="button" value="Texto" (click)="showMapView = false" />
    <agm-map *ngIf="showMapView" style="display:block" [[latitude]="40"
      [[longitude]="2" [[zoom]=5>
      <agm-marker *ngFor="let networkscan of networkscans; let i = index"
        [[latitude]="networkscan.latitude" [[longitude]="networkscan.longitude">
        <agm-info-window>
          <strong>{{networkscan.operator_name}}
            {{networkscan.signal_type}}</strong>
          <p>Intensidad: {{networkscan.signal_classification}}</p>
          <p>{{networkscan.signal_strength}} dB</p>
          <p>(Creado: {{networkscan.created}})</p>
        </agm-info-window>
      </agm-marker>
    </agm-map>
    <table *ngIf="!showMapView" class="table">
      <tbody>
        <tr *ngFor="let networkscan of networkscans; let i = index">
          <td>{{networkscan.operator_name}}
            {{networkscan.signal_type}}</td>
          <td>{{networkscan.signal_strength}} dB
            ({{networkscan.signal_classification}})</td>
          <td>{{networkscan.latitude}}</td>
          <td>{{networkscan.longitude}}</td>
          <td>{{networkscan.created}}</td>
        </tr>
      </tbody>
    </table>
  </div>
</md-sidenav-container>
<footer> <div> © 2018 - Juan Luis Marí Pardo </div> </footer>

```

Aquest és un resum de l'estructura de la pàgina web. Com es pot veure, Angular proporciona la possibilitat d'enllaçar les dades (*data binding*) que ens proporcioni l'API REST directament al codi HTML, per tal que en el moment que arribi la llista de *NetworkScans* de servidor, aquests siguin representats un per un (estan dins la directiva d'Angular *ngFor*, és a dir, que l'estructura de contingut HTML dins d'aquesta directiva es repeteix un cop per cada objecte dins la llista) de manera automàtica a un mapa o a una taula de text, depenent de la variable *showMapView* (degut a la directiva *ngIf* que verifica si la variable és certa o falsa) que canvia en funció dels botons "Mapa" i "Text".

Cal destacar que el mapa (*agm-map*) consta de tants *markers* (*agm-marker*) com *networkscans* rebuts. Mentrestant, cadascun d'aquests *markers* consta amb una vista d'informació (*agm-info-window*) que, un cop es *clicken*, es mostrarà amb tota la informació requerida d'aquell escaneig.

b. L'arxiu CSS que defineix els estils de cadascun dels elements de la pàgina HTML creada, establint-los per:

- i. *Etiqueta.* Per exemple `<agm-map ... />` → `agm-map { // estil }`
- ii. *Classe.* Per exemple `<p class="content">` → `.content { // estil }`
- iii. *Identificador.* Per exemple `<div id="footer">` → `#footer { // estil }`

```

:host {
  display: flex;
  flex: 1;
}

md-sidenav {
  padding-left: 50px;
  padding-right: 50px;
}

.mat-list .mat-list-item,
.mat-nav-list .mat-list-item {
  margin-top: 20px;
}

.content {
  padding: 12px;
}

.fab-bottom-right {
  position: fixed;
  right: 16px;
  bottom: 16px;
}

/deep/ .mat-list-item-content {
  height: auto !important;
}

.selected-subproject {
  color: #d7d0b2;
}

agm-map {
  height: calc(80vh);
  width: calc(100vw - 24px);
}

```

- c. El servei (Service) d'Angular que s'encarregarà de fer la petició HTTP GET a l'API REST per obtenir les dades de xarxes mòbils globals. Aquest estarà lligat amb un model de dades (*NetworkScan*), ja que retornarà de forma asíncrona un llistat de tipus de dada *NetworkScan* (*NetworkScan[]*):

```

import
'rxjs/add/operator/toPromise';

@Injectable()
export class NetworkScanService {

  networkscans: NetworkScan[];

  constructor(private http:
Http) { }

  getNetworkScans() {
    return
this.http.get('http://www.mobil
e-network-scan.com/api/networks
cans/')
    .map(response =>
response.json());
  }
}

export class NetworkScan {
  constructor(
    public id: number,
    public created: string,
    public operator_name:
string,
    public
signal_classification: string,
    public signal_strength:
number,
    public signal_type: string,
    public latitude: number,
    public longitude: number,
  ) { }
}

```

Tots aquest elements han fet possible una arquitectura organitzada de l'aplicació, així com una divisió de funcions per tal de mantenir un codi net i sòlid. De la mateixa manera, la utilització de *Material Design* i de estils variables segons l'altura i amplada, els dissenys són vàlids tant per pantalles grans (monitors d'ordinadors) o per pantalles reduïdes (telèfons mòbils). Els resultats els podem veure a les següents figures, dividits segons diferents dispositius de prova:

Vista d'ordinador (17" i 11")

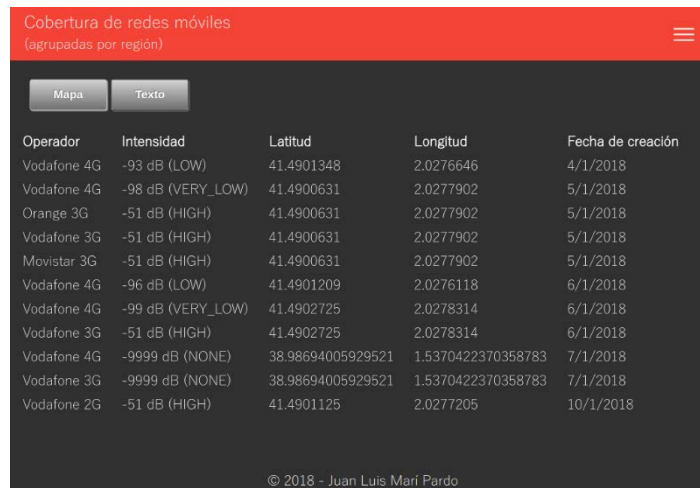
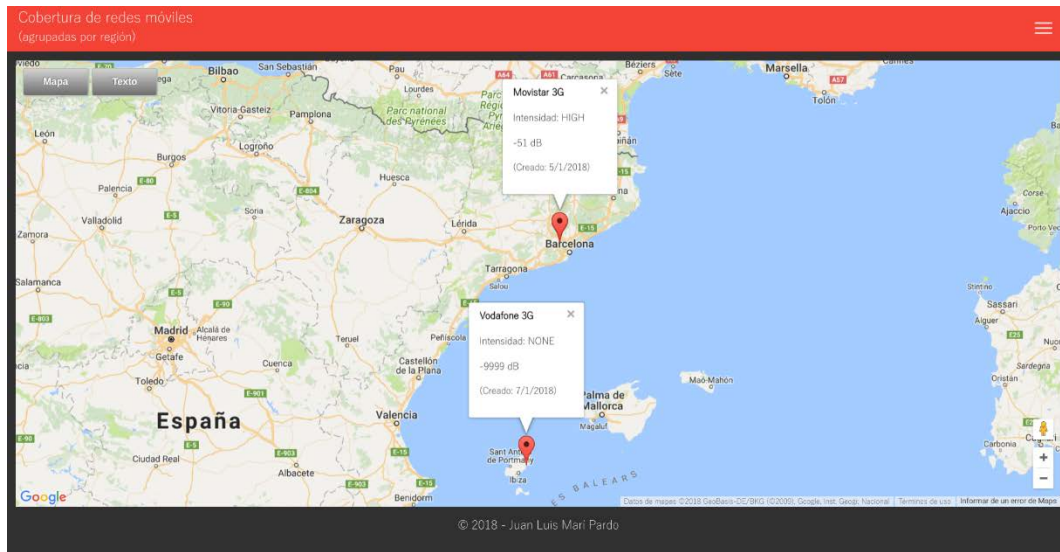


Figura 17. Vistes de la pàgina web a ordinadors de 17" i 11"

Vista mòbil (iPhone 5)

Vista mòbil (Nexus 6)

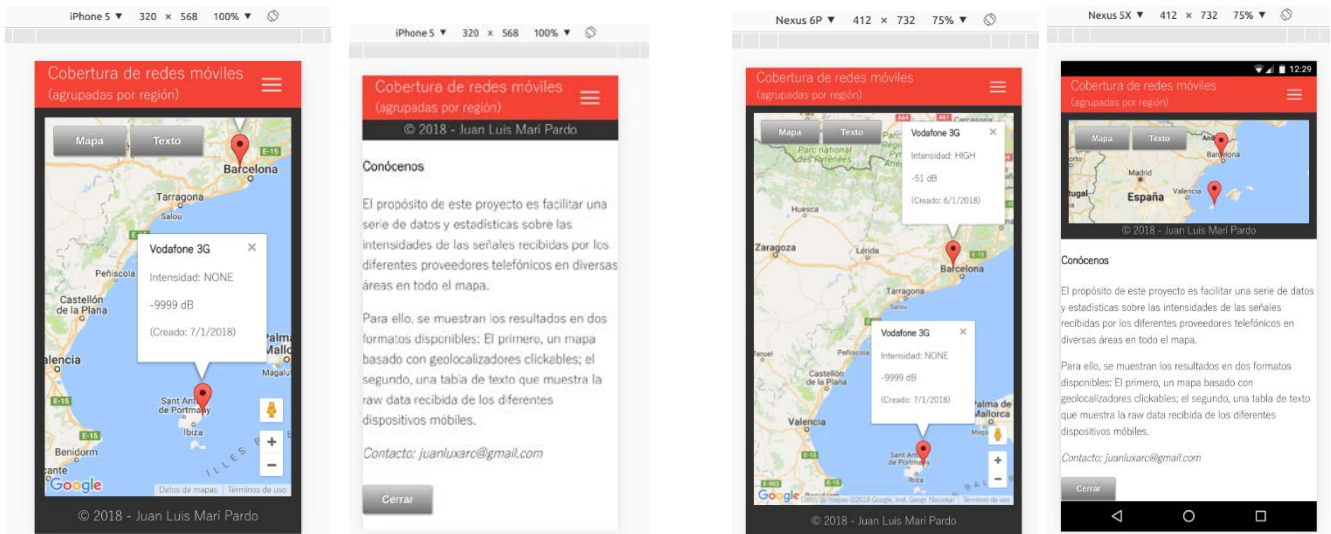


Figura 18. Vistes de la pàgina web a mòbils iPhone 5 i Nexus 6

3. Disseny gràfic i interfícies

Durant el desenvolupament de l'aplicació Android i Web, en tot moment s'ha intentat respectar una sèrie de patrons per tal que ambdues tinguin una correspondència, ja que al cap i a la fi són dues parts d'un mateix producte. Durant aquest punt es concretaran quins han set aquests patrons.

3.1 Estils

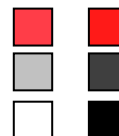
Definició de la línia gràfica del treball. És recomanable incloure, entre altres:

- Logotip. El logotip de l'aplicació Android i l'aplicació Web és el mateix, es tracta d'un símbol que reflecteix la connexió sense fils.



- Paleta de colors. Els colors utilitzats durant el desenvolupament de les dues aplicacions han set els colors "corporatius" (vermell i gris) i els colors "generals" (blanc i negre):

- Vermell. Concretament dues variants d'ell (Iluminosa i fosca):
- Gris. Concretament dues variants d'ell (Iluminosa i fosca):
- Blanc i negre.



```
<color name="dark_grey">#414141</color>
<color name="light_grey">#c0c0c0</color>
<color name="light_red">#ff3d48</color>
<color name="dark_red">#ff1a18</color>
<color name="white">#ffffff</color>
<color name="black">#000000</color>
```

- Paleta tipogràfica. Les tipografies utilitzades han set dues completament distintes a les aplicacions:

- *Android*: Tipografia per defecte, *Sans*.
- *Web*: Tipografia *Yu Gothic*.

- Grandària. En aquest sentit, s'ha tractat de realitzar un treball amb un nombre determinat de mides, cadascun utilitzat segons la seva importància (més gran, més important; i viceversa), dividint els tamany segons "talles":

- *M (mitjà, per defecte)*: 13px
- *L (gran)*: 15px
- *XL (molt gran)*: 25px
- *S (petit)*: 12px
- *XS (molt petit)*: 10px

```
<dimen name="text_xl">25px</dimen>
<dimen name="text_l">15px</dimen>
<dimen name="text_m">13px</dimen>
<dimen name="text_s">12px</dimen>
<dimen name="text_xs">10px</dimen>
```

- Estil de fonts. L'estil de fonts utilitzats ha set el següent, segons la importància del text amb que anaven associats):

- *Regular (per defecte)*
- *Semibold (més importància)*
- *Light/Italic (menys importància)*
- *Bold (títols)*

- Icones: El conjunt de totes les icones que comporten les aplicacions Android i Web (així com el seu logotip) s'ha extret de la pàgina web <https://www.flaticon.es/>, fent les modificacions de colors i ombrejats oportunes per fidelitzar amb els colors corporatius esmentats anteriorment.



3.2 Usabilitat

Per tal de realitzar unes aplicacions captivadores i que millorin el màxim l'experiència d'usuari s'han aplicat una sèrie de principis extrets especialment per Android, Angular i Material, utilitzant en ambdós casos:

- Recordar les respostes de l'usuari. Si no accepta els permisos, el següent ensenyar una alerta explicant per què es requereixen.
- Simplificar amb frases concises i breus.
- Imatges més pràctiques que paraules.
- Preguntar només quan l'acció es consideri necessària. Això ofereix control a l'usuari, però sense molèsties per excessiva càrrega
- Només mostrar la informació necessària quan sigui necessària
- Mantenir informat en tot moment a l'usuari en quin pas es troba i què ha de fer.
- Guardar informació en *cache* o preferències d'usuari per si es produeix un tancament inesperat de l'aplicació.
- Fomentar lo simple i la velocitat.
- Ús de ombres i animacions per donar profunditat i *responsivitat* als dissenys.
- Mantenir un conjunt de colors propis i característics de les aplicacions, per conservar una correspondència entre les diferents vistes i aplicacions.

4. Llenguatges de programació i dependències utilitzades

4.1 Aplicació Android

Pel desenvolupament de l'aplicació Android, el llenguatge de programació triat es tracta de **Kotlin**. Es tracta d'un llenguatge recent, que es va començar a desenvolupar fa sis anys, al gener de 2012, per la companyia *JetBrain* basant-se en la *Màquina Virtual de Java* (en anglès, JVM) per tal de poder ser totalment interoperable amb Java i el seu ecosistema. El dia 17 de maig de 2017, Google va anunciar durant la conferència *Google I/O* que a partir d'aquell moment *Kotlin* es convertiria en el nou llenguatge oficial d'Android pel desenvolupament de les seves aplicacions natives.

- És expressiu, concís, potent. Llenguatge potent alhora de simple d'utilitzar, amb una corba d'aprenentatge lleugera.
- Completament interoperable amb *Java* (anterior llenguatge oficial d'Android).
- És robust. Conté funcions de seguretat per a la nul·litat i la immutabilitat, per fer que les aplicacions d'Android siguin saludables i que funcionin de manera predeterminada.
- És un projecte de codi obert en Apache 2.0.

Les dependències utilitzades i necessàries, així com la seva versió, per l'execució de l'aplicació es poden veure a l'arxiu *build.gradle*. Algunes són necessàries pel funcionament del projecte (*buildTools*, *compileSdk*, *minSdk* i *targetSdk*), altres són necessàries per utilitzar Kotlin i les seves extensions (Kotlin i Anko) i la resta són necessàries per injecció de dependències (Dagger), compressió i descompressió de models de dades (gson) o connexió amb servidor (Retrofit, OkHttp3 i Rx):

```
ext {  
    // Sdk and tools  
    compileSdkVersion = 26  
    buildToolsVersion = '26.0.2'  
    minSdkVersion = 17  
    targetSdkVersion = 26  
  
    // App version  
    versionCode = 4  
    versionName = '1.0.3'  
  
    // App dependencies  
    supportLibraryVersion = '27.0.2'  
    gradleVersion = '3.0.0'  
    kotlinVersion = '1.2.10'  
    ankoVersion = '0.10.1'  
    daggerVersion = '2.5'  
    gsonVersion = '2.8.2'  
    retrofitVersion = '2.3.0'  
    okHttpVersion = '3.9.0'  
    rxKotlinVersion = '2.1.0'  
    rxAndroidVersion = '2.0.1'  
  
    // Unit tests dependencies  
    junitVersion = '4.12'  
  
    // Instrumentation tests dependencies  
    espressoVersion = '2.2.2'  
}
```


4.2 Sistema Back-End

Pel desenvolupament del sistema Back-End, el llenguatge de programació triat es tracta de **Django**. Es tracta d'un llenguatge de desenvolupament de codi obert, desenvolupat el juliol de 2005 per Lawrence Journal-World (i companys de l'empresa *World Online*), punter en el desenvolupament de software per servidors web. Els seus avantatges principals són:

- Es tracta d'un framework d'alt nivell de *Python*.
- Reforça un desenvolupament ràpid i net, amb un disseny pragmàtic. Segueix el Model-Vista-Controlador (MVC) per facilitar els dissenys en l'arquitectura de les aplicacions i separar funcionalitats en diferents mòduls, de manera que els testos siguin molt més senzills d'implementar i poder mantenir un codi net i robust.
- Estructura bàsica, simple i organitzada:
 - *Routers (enrutadors)*: Eina per definir les direccions URI de l'API a les quals s'hauran de fer les peticions web pertinents.
 - *Views (vistes)*: Eina per gestionar el flux des de que es rep la petició del client fins que es genera la resposta del servidor.
 - *Serializers (serialitzadors)*: Instruments que s'encarreguen de transformar les dades de la petició i processar-les de la manera apropiada segons el contingut i tipus de petició.
- Separació entre client i servidor. La interfície que ofereix aïlla l'usuari de cap coneixement sobre la base de dades (tipus d'estructura, servidor on s'allotja, taules disponibles) que emmagatzema la informació.
- Visibilitat i fiabilitat millorades. Aquesta via ofereix una gran escalabilitat per tal de portar el desenvolupament a un servidor diferent, ampliar el nombre de servidors connectats per tal de gestionar millor els processos i recursos amb balancejadors de càrrega, i gestió organitzada i estructurada de la informació.
- Independència del llenguatge i plataformes utilitzades en la resta de codi. Això permet el seu accés des de qualsevol tipus d'ambient de treball o *environment*, és a dir, que qualsevol actualització del llenguatge a un de nou utilitzat per sol·licitar recursos no suposarà cap canvi en la arquitectura ni el codi d'aquesta part. Per tant, es podria portar el llenguatge o ampliar-ho sense cap tipus de problema.

Les dependències utilitzades i necessàries, així com la seva versió, per l'execució de l'aplicació es poden veure a l'arxiu *requirements.txt*. Són necessàries per utilitzar Django com API REST i per permetre intercanvi de recursos d'origen creuat (CORS). No obstant, al tractar-se d'un *framework* de Python, també es requereix la versió de *Python 2.7* així com el mòdul "*pip*", per la seva compilació i instal·lació de dependències respectivament:

```
Django==1.11
djangorestframework==3.7.3
django-cors-headers==2.1.0
```

4.3 Aplicació Front-End

Pel desenvolupament de l'aplicació Front-End, el llenguatge de programació triat es tracta de **Angular2**. Es tracta d'un llenguatge de codi obert, llançat el 15 de setembre de 2016 per *Google* i que representa l'evolució d'*AngularJS* (la primera versió del llenguatge), però que no consta d'interoperabilitat amb ell degut als seus majors canvis i funcionalitats totalment noves:

- Es tracta d'un *framework* de *Javascript*, basat en *TypeScript*.
- Segueix el Model-Vista-Controlador (MVC) per facilitar els dissenys en l'arquitectura de les aplicacions i separar funcionalitats en diferents mòduls, de manera que els testos siguin molt més senzills d'implementar i poder mantenir un codi net i robust.
- És multiplataforma. Tot i la seva orientació a la creació d'aplicacions web, també s'utilitza comunament per crear aplicacions d'escriptori (*Windows, Mac, Linux*) o per crear aplicacions mòbils híbrides combinant amb altres plataformes com *Cordova i Ionic*.
- Alta velocitat i rendiment de generació de codi i de divisió del mateix (gràcies a la seva combinació amb *Webpack*).
- Altament testeable, mitjançant la seva combinació amb llibreries de *Karma i Protractor* per realitzar testos unitaris i d'integració de forma simple i a un entorn aïllat.
- Presenta un IDE intuïtiu, integrat amb diferents eines d'instal·lacions de dependències, com ara bé *Angular CLI i npm*.

Les dependències utilitzades i necessàries, així com la seva versió, per l'execució de l'aplicació es poden veure a l'arxiu *package.json*. Es requereix l'instal·lador de paquets *npm* per poder integrar-les com a mòduls de l'aplicació:

```
"dependencies": {
  "@agm/core": "^1.0.0-beta.2",
  "@angular/animations": "4.2.5",
  "@angular/cdk": "2.0.0-beta.8",
  "@angular/common": "4.2.5",
  "@angular/compiler": "4.2.5",
  "@angular/core": "4.2.5",
  "@angular/flex-layout": "2.0.0-beta.8",
  "@angular/forms": "4.2.5",
  "@angular/http": "4.2.5",
  "@angular/material": "2.0.0-beta.8",
  "@angular/platform-browser": "4.2.5",
  "@angular/platform-browser-dynamic":
"4.2.5",
  "@angular/router": "4.2.5",
  "core-js": "2.4.1",
  "hammer": "0.0.5",
  "hammerjs": "2.0.8",
  "rxjs": "5.3.0",
  "zone.js": "0.8.12"
},
"devDependencies": {
  "@angular/cli": "1.2.0",
  "@angular/compiler-cli": "4.2.5",
  "@types/jasmine": "2.5.46",
  "@types/node": "7.0.12",
  "codemirror": "3.0.1",
  "simple-git": "1.69.0",
  "simple-terminal-menu": "1.1.3",
  "ts-node": "3.0.4",
  "tslint": "5.3.2",
  "typescript": "2.3.4"
}
```

Tot i semblar moltes dependències, cal esmentar que totes les dependències que comencen per *@angular/** són requerides pel funcionament de l'aplicació. Mentre que la resta es tracten de dependències opcionals, a excepció de "rxjs" que servirà per la realització de les trucades a l'API REST. En general s'utilitzaran per millorar l'experiència com a desenvolupador (*core, hammer, zone*), permetent la integració amb repositoris de GIT (*simple-git*), afegint dissenys de Material o d'altres terminals, anàlisi de la integritat del codi (*codemirror*) i automatització de testos (*jasmine*).

Capítol 4: Implementació

1. Requisits d'instal·lació

En aquest punt s'explicarà quins són els requisits necessaris a nivell de software i hardware per instal·lar qualsevol de les aplicacions creades.

- **Hardware:**
 - Ordinador amb Sistema Operatiu (Windows, Ubuntu o Mac).
 - Exclusivament per la compilació de l'aplicació Android també s'hauria de disposar d'un mòbil amb Android de versió API 17 o major.
- **Software:**
 - Per tal de compilar l'aplicació Android: *Android Studio v.3.0* o major. Instal·lar dependències de l'arxiu *build.gradle* (automàticament per l'IDE).
 - Per tal de compilar l'aplicació Back-End: *Python v.2.7*, mòdul "*pip*". Instal·lar dependències de l'arxiu *requirements.txt* amb "*pip*". Es recomana instal·lar l'IDE *Python Community Edition*.
 - Per tal de compilar l'aplicació Front-End: Gestor de paquets "*npm*". Instal·lar dependències de l'arxiu *package.json* amb "*npm*".

Al següent punt es pot veure detalladament els passos d'instal·lació de cadascuna d'aquestes tecnologies per tal de poder replicar amb exactitud el procediment original utilitzat pel desenvolupament d'aquest projecte a data actual.

2. Instruccions d'instal·lació

En cas que es vulgui instal·lar alguna de les aplicacions en local, s'hauran de realitzar diferents passos segons la tecnologia implicada:

1. **Android:** Instal·lar una versió igual o superior a *Android Studio v3.0* (compatible amb *Kotlin*) i obrir el projecte enviat (*mobile-network-scan.zip*). Automàticament s'instal·laran les dependències de l'arxiu *build.gradle*, un cop fet, click a "*Build project*" i finalment "*Run*".
 - o Selecció de l'opció "*build variant*", es pot canviar l'entorn de l'aplicació de "*debug*" (tests) a "*release*" (producció).
2. **Back-End:** Instal·lar *Python v2.7* juntament amb "*pip*".
 - a. OPCIÓ 1: Des de *pip*, instal·lar manualment totes les dependències de l'arxiu *requirements.txt*, dins el projecte enviat (*network-scan-database.zip*) mitjançant la comanda:
 - i.

```
pip install -r requirements.txt
```Executar l'arxiu *manage.py*, mitjançant la comanda:
 - ii.

```
python manage.py runserver
```
 - b. OPCIÓ 2: Instal·lar *PyCharm Community Edition* i obrir el projecte enviat (*network-scan-database.zip*), que instal·larà automàticament totes les dependències de l'arxiu *requirements.txt*. Executar l'arxiu *manage.py* des de l'opció de "*Run*".
 - c. Veure els canvis des d'un navegador a la URL: <http://localhost/api/networkscans>
3. **Front-End:** Instal·lar el gestor de dependències "*npm*" i "*node-sass*".
 - a. Des de l'arrel del projecte enviat (*coverage-network-scan-region-gruped.zip*), executar la comanda que instal·larà totes les dependències de l'arxiu *package.json*:
 - ```
npm install
```
  - b. Un cop tenint les dependències, executar el servidor local amb l'aplicació web amb la comanda:
    - ```
npm start
```
 - c. En cas de voler crear els fitxers estàtics de l'entorn de producció, es podran trobar a la carpeta auto-generada anomenada "*dist*" després d'executar la comanda:
 - ```
npm run prodbuild
```

Cal observar que en totes tres tecnologies s'ha creat un arxiu que recopila el conjunt de dependències necessàries. Això s'ha fet per facilitar la instal·lació i l'ús del projecte:

- A *Android* s'utilitza *gradle* mitjançant l'arxiu *build.gradle*
- A *Python* (i per tant, *Django*) s'utilitza *pip* mitjançant l'arxiu *requirements.txt*
- A *JavaScript* (i per tant, *Angular2*) s'utilitza *npm* mitjançant l'arxiu *package.json*

# Capítol 5: Demostració

## 1. Instruccions d'ús

Per utilitzar el sistema creat, l'únic que s'ha de fer és:

1. Instal·lar l'APK enviat (*mobile-network-scan.apk*)
2. Seguir les instruccions d'ús de l'aplicació (pantalla ajuda) per tal d'escanejar dades de xarxes mòbils. Un cop creades les dades, l'aplicació permet compartir-les amb la base de dades utilitzant l'opció de la *toolbar* des de la pantalla de resultats
3. En aquest moment, per verificar que les dades s'han enviat correctament, es pot entrar a la pàgina web del *back-office*: <https://www.mobile-network-scan.com/api/networkscans> i verificar segons la data d'inserció (el darrer "*id*" [identificador] es correspon amb la darrera inserció)
4. Finalment, per visualitzar la cobertura del conjunt total de xarxes d'operadors mòbils agrupades per regió, entrar a la pàgina web de l'aplicació *front-end*: <http://www.mobile-network.com>

## 2. Tests

Durant el desenvolupament del projecte s'han realitzat diversos tests per tal d'assegurar la correcta funcionalitat sense errors i segura del sistema. Per fer-ho, s'han realitzat diverses variacions de tests, segons les diferents tecnologies emprades.

### 2.1 Aplicació Android

Per tal de verificar que l'aplicació Android funciona de la manera adequada, s'han realitzat una sèrie de tests automatitzats a nivell d'integració i altres d'unitaris. És a dir, s'han realitzat tests que verifiquen la correcta funcionalitat de parts específiques de l'aplicació, i altres que verifiquen la correcta funcionalitat de procediments dins de l'aplicació. Tots ells comprenen aspectes d'usuari i usabilitat, i així com l'aplicació, han estat escrits amb llenguatge *Kotlin*:

```
class HelpActivityTests {

 @get: Rule
 val helpActivityRule = ActivityTestRule(HelpActivity::class.java)

 @Test
 fun toolbar_isDisplayed() {
 onView(isAssignableFrom(Toolbar::class.java)).check(matches(isDisplayed()))
 }

 @Test
 fun toolbar_hasCorrectText() {
 onView(isAssignableFrom(Toolbar::class.java)).check(matches(withToolbarTitle(
 _is(helpActivityRule.activity.getString(R.string.help_title))))))
 }

 @Test
 fun pressUpButton_finishActivity() {
 onView(homeAsUpButton()).perform(click())
 assertTrue(helpActivityRule.activity.isFinishing)
 }

 @Test
 fun pressBack_finishActivity() {
 try {
 pressBack()
 } catch (e: NoActivityResumedException) {
 assertTrue(helpActivityRule.activity.isFinishing)
 }
 }
}

class HomeActivityTests {

 @get: Rule
 val homeActivityRule = ActivityTestRule(HomeActivity::class.java)

 @Before
 fun setUp() {
 removeSharedPreferences()
 Intents.init()
 }

 @After
```

```

 fun tearDown() {
 Intents.release()
 }

 @Test
 fun toolbar_isDisplayed() {
 onView(isAssignableFrom(Toolbar::class.java)).check(matches(isDisplayed()))
 }

 @Test
 fun toolbar_hasCorrectText() {
onView(isAssignableFrom(Toolbar::class.java)).check(matches(withToolbarTitle(
 _is(homeActivityRule.activity.getString(R.string.home_title))))))
 }

 @Test
 fun startButton_isClickable() {
 onView(withId(R.id.btnScan)).check(matches(isClickable()))
 }

 @Test
 fun helpButton_isClickable() {
 onView(withId(R.id.btnHelp)).check(matches(isClickable()))
 }

 @Test
 fun onHelpButtonClick_navigatesToHelpActivity() {
 onView(withId(R.id.btnHelp)).perform(click())
 intended(hasComponent(HelpActivity::class.java.name))
 }

 @Test
 fun onScanButtonClick_navigatesToSystemPhonePackage() {
 onView(withId(R.id.btnScan)).perform(click())
 intended(toPackage("com.android.phone"))
 }

 @Test
 fun pressBack_finishActivity() {
 try {
 pressBack()
 } catch (e: NoActivityResumedException) {
 assertTrue(homeActivityRule.activity.isFinishing)
 }
 }

 @Test
 fun pressBack_finishApp() {
 try {
 pressBack()
 } catch (e: NoActivityResumedException) {
 // Success
 }
 }

 private fun removeSharedPreferences() {
 val memoryDataSource = MemoryDataSource(homeActivityRule.activity,
 MemoryDataSource.PreferencesType.PREFS_USER)
 if (memoryDataSource.getNetworkOperatorList()?.size ?: 0 > 0) {
 MemoryDataSource(homeActivityRule.activity,
 MemoryDataSource.PreferencesType.PREFS_USER)
 .removeNetworkOperatorList()
 pressBack()
 }
 }
}

```

```

class ScanResultsActivityTests {

 @get: Rule
 val scanResultsActivityRule = ActivityTestRule(ScanResultsActivity::class.java)

 @Test
 fun toolbar_isDisplayed() {
 onView(isAssignableFrom(Toolbar::class.java)).check(matches(isDisplayed()))
 }

 @Test
 fun toolbar_hasCorrectText() {

onView(isAssignableFrom(Toolbar::class.java)).check(matches(withToolbarTitle(
_is(scanResultsActivityRule.activity.getString(R.string.scan_results_title))))))
 }

 @Test
 fun pressUpButton_finishActivity() {
 onView(homeAsUpButton()).perform(click())
 assertTrue(scanResultsActivityRule.activity.isFinishing)
 }

 @Test
 fun pressBack_finishActivity() {
 try {
 pressBack()
 } catch (e: NoActivityResumedException) {
 assertTrue(scanResultsActivityRule.activity.isFinishing)
 }
 }
}

```

Tots aquests tests funcionen correctament.

Cal notar que cada cop que es realitza un conjunt de tests al finalitzar-los es realitzen, si cal, les funcions necessàries per mantenir l'entorn de l'aplicació net i lliure, és a dir, els tests es realitzen en un entorn tancat i que no afecti a l'aplicació en el seu mode de producció (per exemple, la funció `removeSharedPreferences`).

Finalment, s'ha verificat la integritat del codi establint un entorn de producció que minifica i ofusca el codi realitzat, mitjançant l'eina *proguard*.

## 2.2 Sistema Back-End

Sobre el sistema *back-end*, cal esmentar que la mateixa API de Django autogestiona els errors ocasionats per falta de connectivitat, time-out, o problemes interns del servidor, retornant diferents codis segons l'error o si tot ha anat correctament. Per tal de verificar que el sistema Back-End respon correctament a les peticions HTTP GET i POST que permeten obtenir el conjunt de resultats d'escaneig o afegir de nous, respectivament, s'ha utilitzat un sistema de *back-office* que el mateix llenguatge *Django* ofereix. Es pot trobar o <http://www.mobile-network-scan.com/api/networkscans/>, i permet diverses accions sobre l'*endpoint* generat pel llistat d'escàners de xarxa mòbil:

- En primer lloc, ofereix els resultats de la petició GET en formats JSON i API.



- En segon lloc, permet realitzar peticions POST de prova i veure de forma simultània els seus efectes sobre la base de dades o els errors que pot ocasionar.
- Finalment, es poden veure les opcions (OPTIONS) del tipus de contingut generat.

### 2.3 Aplicació Front-End

L'aplicació Front-End ha set testejada de dues maneres distintes, sense automatitzacions:

- En primer lloc, s'ha realitzat una sèrie de testos des de la consola del navegador Chrome, *debugant* diverses parts del codi en entorn local i de reproducció. Per tant, es pot tractar com testos unitaris sobre la lògica de l'aplicació, però manuals.
- En segon lloc, s'ha realitzat una sèrie de testos *end-to-end* manuals, és a dir, s'han anat provant les diferents interaccions amb el contingut de la pàgina web esperant que fessin l'acció oportuna:
  - *Click al botó de "Mapa"*:
    - Obre el mapa en cas de no estar prèviament obert
    - No fa res en cas de estar prèviament obert
  - *Click al botó de "Text"*:
    - Obre la taula de text en cas de no estar prèviament oberta
    - No fa res en cas de estar prèviament oberta
  - *Ampliar/reduir el mapa*:
    - Augmenta/reduïx respectivament el zoom del mapa
  - *Clickar a la icona "menú hamburguesa"*:
    - Mostra l'opció d'informació del projecte, en cas de no estar prèviament oberta
    - Tanca l'opció d'informació del projecte en cas d'estar prèviament oberta
  - *Click a l'opció d'informació del projecte*:
    - Obre un diàleg modal que ofereix informació del projecte
  - *Click al botó de tancar informació del projecte*:
    - Tanca el diàleg d'informació del projecte

Finalment, s'ha verificat la integritat del codi establint un entorn de producció que minifica i ofusca el codi realitzat i que genera un conjunt de fitxers estàtics a publicar a la pàgina web de l'aplicació.

# Capítol 6: Conclusions i línies de futur

## 1. Conclusions

Durant el desenvolupament d'aquest treball s'ha treballat la conceptualització de desenvolupaments de *software* d'una manera sòlida i robusta, independentment del llenguatge de programació. Aquest concepte ha set especialment útil per després la realització dels diferents programes en que es basa el producte sencer, ja que és aplicable tant a desenvolupament d'aplicacions en Android com pel desenvolupament d'aplicacions per Back-End i Front-End. Quant a la realització de les aplicacions, ha set un projecte molt productiu, ja que ha permès aprendre els llenguatges de programació *Kotlin* per Android, llenguatge amb molts avantatges en front Java, *Django* per Back-End, i *Angular2* per Front-End; que amb gran seguretat serviran per futurs projectes i treballs. A més, s'han après altres conceptes importants com la necessitat d'un balancejador de càrrega que permeti prevenir l'excés de peticions al servidor, la creació d'aquest servidor i enllaçat amb un domini web, i s'ha aprofundit en la teoria sobre màrqueting i monetització de l'aplicació per futures versions.

Els objectius generals del projecte eren la captació de dades sobre cobertura de xarxes mòbils mitjançant un dispositiu mòbil i la possibilitat de veure el conjunt de totes les dades preses a una pàgina web intuïtiva. D'aquesta manera, es pot dir que els objectius sí han estat assolits, gràcies possiblement a treballar amb marges de temps i d'error que s'han anat convertint en essencials i a la metodologia de seguiment utilitzada per verificar que el flux de treball es complia amb la velocitat adequada. Sobre aquest tema, l'únic detall negatiu que es pot remarcar és que la pàgina web podria ser molt més útil i intuïtiva si tingués opcions de filtrat i ordenació segons algun camp del model de dades, però tot i tractar-se d'una primera versió, ja compta amb la visualització en diversos formats: mapa i taula de text, que era un objectiu secundari però primordial alhora de mostrar la informació de manera més clarivident i intuïtiva.

A nivell de metodologia de treball i de desenvolupament del mateix ha set un repte important treballar amb metodologia àgil, seguint *Scrum*, ja que requereix tenir consciència en tot moment de quina és la divisió en tasques del producte sencer, i d'estimar amb la major claredat possible quines són els desenvolupaments prioritaris i quins són els que requereixen un esforç major o menor. Es pot veure que aquest mètode un cop s'utilitza a nivell pràctic és molt útil pel desenvolupament de projectes que poden variar respecte la seva forma inicial, degut a la seva dinàmica, però que és un mètode que es basa en l'aprenentatge: quant més l'utilitzes, millor estimes les tasques i més fàcil és arribar a complir els propòsits de cada Sprint i de cada entregable. En aquest projecte l'avanç és clar, durant la primera fase del projecte el retard es va donar de 11 dies, degut a la poca experiència planificant i estiment tasques; mentre que al segon i al tercer, un cop ja es tenien en compte més factors alhora d'estimar tasques, només es va produir un retard de 4 dies. En total, s'han produït 19 dies de retard.

Per tant, la metodologia Scrum ha set molt funcional i ha servit aquest projecte per tal de modificar breument la planificació segons el projecte avançava i es consolidaven els objectius, però sempre

respectant uns objectius finals que s'han aconseguit assolir dins el marge màxim d'error que s'havia establert. D'aquests canvis, només hi calen destacar les dificultats per captar informació necessària de xarxes i detecció de canvis de senyal a l'aplicació Android (especialment amb operadors diferents del contractat) i la inexperiència amb llenguatges de *back-end* i el seu entorn de desenvolupament.

Finalment, quant als resultats obtinguts dels escàners s'han extret conclusions molt interessants. La primera és que les connexions de generacions anteriors de xarxes mòbils obtenen una millor cobertura que les connexions 4G o 4G+, la qual cosa es pot comprovar fàcilment a zones on la connexió 4G no arriba, ja que el propi telèfon es connecta llavors a les xarxes 3G o 2G amb aquest ordre de prioritat, però que sembla que es podrà confirmar de forma definitiva quan s'obtinguin resultats d'escàners d'un públic geogràficament distribuït. La segona va molt relacionada amb aquesta, doncs, tot i les millores en velocitat i latència de les noves xarxes mòbils la intensitat de la senyal encara es pot millorar de forma considerable, pel que es podria arribar a pensar si és millor opció continuar treballant en augmentar la velocitat amb la nova generació 5G (esmentat en el capítol 2.1), o si la millor opció és intentar millorar la infraestructura 4G per rebre millor connectivitat, o una combinació de totes dues.

## 2. Línies de futur

El producte entregat es tracta d'una primera versió d'un servei que pretén continuar desenvolupant-se en el futur. Després d'extreure unes conclusions sobre aquesta versió, també es poden extreure una sèrie de línies de futur per tal d'incrementar les seves possibilitats d'èxit:

En primer lloc, un aspecte molt important a remarcar són els dissenys de les aplicacions, en aquest cas, s'hauria de fer un treball d'anàlisi per descobrir si els usuaris els troben intuïtius, si són dissenys agradables, i si la seva influència és positiva o negativa alhora d'utilitzar les aplicacions d'Android i Web.

En segon lloc, es tractaria de millorar els aspectes de l'aplicació web en relació al mostreig de les dades obtingudes, ja que el mapeig és una bona idea, però en versions futures la intenció és que els usuaris tinguin un major control de les dades que estan mostrant-se tant en el format mapa com text, podent filtrar en ambdós casos segons algun dels seus camps (companyia, tipus de senyal, intensitat, data de creació) o podent ordenar les dades de text segons algun dels camps esmentats.

Finalment, un cop finalitzats els aspectes de disseny i d'usabilitat de les aplicacions, la darrera línia de treball s'hauria de basar en una forma assequible de vendre el producte als usuaris, així com les relacions amb els diferents *partners* i les mateixes companyies telefòniques; aspecte que ja s'ha tractat a l'apartat 2 del capítol 2 de forma teòrica però que encara no s'ha portat a terme en aspectes pràctics.

## Bibliografia

- [1] **Introducción a los smartphones**. <https://histinf.blogs.upv.es/2012/12/03/smartphones/>, consultat 03/01/2018
- [2] **Number of smartphone users world wide from 2014 to 2020 (in billions)**. <https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/>, consultat 08/01/2018
- [3] **Current World Population**. <http://www.worldometers.info/world-population/>, consultat 05/01/2018
- [4] **NewsBizBlog, from the world news media network** (08 Juny 2015). <https://newsbizblog.org/2015/06/08/spain-uk-top-europes-smartphone-penetration-list/>, consultat 10/01/2018.
- [5] **Guowang Miao; Jens Zander; Ki Won Sung; Ben Slimane** (2016). *Fundamentals of Mobile Data Networks*. Cambridge University Press. ISBN 1107143217.
- [6]. **Casuso, Manuel** (29 Juny 2017). *iOS vs Android, tendencias y cuota de mercado*. <https://lextrend.com/ios-vs-android-tendencias-y-cuota-de-mercado/>
- [7] **Parkinson, Spilker** (1996). *Global Positioning System Theory and Applications*. American Institute of Aeronautics and Astronautics. [https://books.google.es/books?id=lv11a5J\\_4ewC&redir\\_esc=y](https://books.google.es/books?id=lv11a5J_4ewC&redir_esc=y)
- [8] **Black Duck Software. MySQL: Project Summary. Openhub**. <https://www.openhub.net/p/mysql>, consultat 05/01/2018
- [9] **Hamilton, Coman** (28 Octubre 2014). *A sneak peek at the radically new Angular 2.0*. <https://jaxenter.com/angular-2-0-112094.html>
- [10] **Android Developer Official WebSite**. <https://developer.android.com/about/dashboards/index.html?hl=es-419>, consultat 10/10/2017
- [11] **Sandi Metz** (Maig 2009). *SOLID Object-Oriented Design*. Duke University, Gotham Ruby Conference
- [12] **Kumar, Liu, Sengupta, and Divya, Vol. 1, Issue 1** (14 Decembre 2010). *Evolution of Mobile Wireless Communication Networks: 1G to 4G*. International Journal on Electronics & Communication Technology (IJECT), pp. 68-72, ISSN 2230-7109
- [13] **Osseiran, A.; Boccardi, F.; Braun, V.; Kusume, K.; Marsch, P.; Maternia, M.; Queseth, O.; Schellmann, M.; Schotten, H.** (1 Maig 2014). *Scenarios for 5G mobile and wireless communications: the vision of the METIS project*. IEEE Communications Magazine. 52 (5): 26–35. doi:10.1109/MCOM.2014.6815890. ISSN 0163-6804.
- [14] **OpenSignal.com** (3 Agost 2017). *Speedtest i mapes 3G 4G WiFi*. <https://play.google.com/store/apps/details?id=com.staircase3.opensignal>
- [15] **KAIBITS Software GmbH** (13 Septembre 2017). *Informació de senyal de xarxa*. <https://play.google.com/store/apps/details?id=de.android.telnet>
- [16] **Test de velocidad**. <https://www.testdevelocidad.es/>, consultat 07/01/2018

## Annex A: Glossari

*API: Application Programming Interface. Interfície de programació d'aplicacions.* Es tracta del conjunt de biblioteques que ofereixen accés als serveis des dels processos i representa un mètode per abstracció a la programació, establint diverses capes de programació.

*CSS: Cascading Style Sheet. Fulla d'estils en cascada.* És un llenguatge de disseny gràfic per tal de definir i crear la presentació d'un document estructurat escrit en un llenguatge de marcat, com per exemple, *HTML*.

*CDMA: Code Division Multiple Access. Mètode de multiplexació per divisió de codi amb múltiple accés.*

*CORS: Cross-Origin Resource Sharing. Intercanvi de recursos d'origen creuat.* És un mecanisme que utilitza capçaleres *HTML* addicionals per permetre que un usuari (*user-agent*) obtingui permisos per accedir a recursos seleccionats des d'un servidor d'origen distint al que pertany.

*FDMA: Frequency Division Multiple Access. Mètode de multiplexació per divisió de freqüència amb múltiple accés.*

*GSM: Global System for Mobile Communications. Sistema global per les comunicacions mòbils.* És un estàndard de telecomunicacions mòbils basat en l'objectiu de poder transmetre missatges curts de text que actualment es coneix com 2G.

*GPS: Global Positioning System. Sistema de posicionament global.* És un sistema que permet ubicar la posició d'un element dins la totalitat de la Terra amb una precisió de pocs metres, mitjançant una sincronització entre satèl·lits que envolten el planeta.

*HTML: HyperText Markup Language. Llenguatge de marques d'hipertext.* És un llenguatge de marcat estàndard per l'elaboració de pàgines web. Serveix per definir el contingut i el posicionament de diversos elements dins l'estructura de la pàgina, com per exemple, imatges, texts, vídeos.

*HTTP: HyperText Transfer Protocol. Protocol de transferència d'hipertext.* És el protocol de comunicació a través del qual es realitzen transferències d'informació en la *World Wide Web (WWW)*.

*PDA: Personal Digital Assistant. Asistent personal digital.* Es tracta de qualsevol dispositiu computacional de ma, com per exemple, telèfons intel·ligents o agendes digitals.

*PSTN: Public Switched Telephone Network. Red telefònica pública commutada.* Es defineix com a un conjunt d'elements constituïts per tots els medis de transmissió i commutació necessaris per enllaçar dos terminals mitjançant un circuit físic.

*REST: Representational State Transfer. Transferència d'estat representacional.* És un estil d'arquitectura de software per sistemes d'hipermedia que compren qualsevol sistema que utilitzi el protocol *HTTP* per modificar dades (obtenir, enviar, esborrar, actualitzar).

*SMS: Short Message Service. Servei de missatges curts.* És un servei disponible a telèfons mòbils que permet enviar missatges de text curts com a part de l'estàndard *GSM*.

*SQL: Structured Query Language. Llenguatge de consulta estructurada.* És un llenguatge específic del domini d'un sistema de base de dades relacionals que permet especificar diverses operacions damunt aquest. Es basa en l'àlgebra i càlcul relacional per realitzar les seves consultes i augmentar la velocitat de cerca d'informació dins la base de dades.

*TDMA: Time Division Multiple Access. Mètode de multiplexació per divisió de temps amb múltiple accés.*

*URL: Uniform Resource Locator. Localitzador uniforme de recursos.* És un identificador únic que apunta a un servidor per un conjunt de recursos variants en el temps.

*WLAN: Wireless Local Area Network. Red d'àrea local sense fils.* És un sistema de comunicació sense fils per minimitzar les connexions cablejades, mitjançant ones de ràdio per portar a terme el traspàs de la informació sense un medi guiat.

*WWW: World Wide Web. Red informàtica mundial.* És un sistema de distribució de documents interconnectats d'hipertext accessible mitjançant Internet, a partir d'un navegador web.

## Annex B: Lliurables del projecte

1. Instruccions per utilitzar el projecte compilat
  - a. APK d'Android
  - b. Fitxer de text amb les URL de les aplicacions Back-End i Front-End
2. Informe de treball del tercer lliurament en format PDF
3. Arxius de treball, dividits en diferents carpetes segons tecnologia i amb instruccions d'ús:
  - a. Android – Kotlin
  - b. Back-End – Django
  - c. Front-End – Angular
4. Memòria en format PDF (aquest document)
5. Presentació acadèmica (vídeo – Present@)
6. Presentació pública (PowerPoint) en format PDF

## Annex C: Currículum Vitae



*Juan Luis Marí Pardo*

*Eivissa, Illes Balears, España. 1993*

*Programador en Visual Engineering*

*Universitat Politècnica de Catalunya - Universitat Oberta de Catalunya*

Graduat en Enginyeria de Telecomunicacions en l'especialitat de Sistemes Audiovisuals per la UPC i realitzant el Treball Fi de Màster d'Aplicacions Multimèdia a la UOC.

Interessat en el món de la programació i el desenvolupament d'aplicacions. Vaig començar la meua carrera professional l'any 2015 fins a dia d'avui, portant a terme diverses aplicacions web i mòbils: *Deporvillage, Metro Madrid, Massimo Dutti, Securitas, Labs. Esteve, Oonair*. Contribucions diverses a *Pyrebase, GstreamerCodeSnippets, Janus-Gateway i Sharepoint Exchange*.