



AlarmMote:

Aplicació d'alarmes sincronitzades en diversos dispositius Android.

Francesc Estrades Juan

Grau en Enginyeria Informàtica

Desenvolupament d'aplicacions per a dispositius mòbils (Android)

Joan Vicent Orenge Serisuelo i Jordi Almirall López

Carles Garrigues Olivella

06/2019



Aquesta obra està subjecta a una llicència de [Reconeixement-NoComercial-SenseObraDerivada 3.0 Espanya de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FITXA DEL TREBALL FINAL

Títol del treball:	<i>AlarmMote: Aplicació d'alarmes sincronitzades en diversos dispositius Android.</i>
Nom de l'autor:	<i>Francesc Estrades Juan</i>
Nom del consultor/a:	<i>Joan Vicent Orenga Serisuelo i Jordi Almirall López</i>
Nom del PRA:	<i>Carles Garrigues Olivella</i>
Data de lliurament (mm/aaaa):	<i>06/2019</i>
Titulació o programa:	<i>Grau en Enginyeria Informàtica</i>
Àrea del Treball Final:	<i>Desenvolupament d'aplicacions per a dispositius mòbils (Android)</i>
Idioma del treball:	<i>Català</i>
Paraules clau	<i>Alarma, Android, Firebase.</i>
Resum del Treball (màxim 250 paraules): <i>Amb la finalitat, context d'aplicació, metodologia, resultats i conclusions del treball</i>	
<p>Cada vegada més els usuaris disposen de diversos dispositius electrònics amb diferents serveis sincronitzats entre ells, mitjançant les xarxes de comunicacions. Des d'un d'aquests dispositius es pot accedir al contingut dels altres i viceversa ja que les dades que comparteixen aquests dispositius realment es troben al "núvol", o sigui distribuïdes en servidors i accessibles remotament mitjançant Internet.</p> <p>Entre els diversos serveis possibles, un d'ells podria ser un sistema d'alarmes amb capacitats extra associades a aquestes. Una aplicació amb aquestes característiques posseiria d'un caràcter transversal pel gran conjunt d'usos que se'n podrien fer servir en molts camps i contextos diferents.</p> <p>Per l'elaboració del treball s'han utilitzat diverses metodologies tant en la planificació, passant pel disseny centrat en l'usuari en quan a la interacció amb l'usuari i en el disseny tècnic de l'estructura de l'aplicació així com també del model base de dades i finalment la implementació del producte utilitzant metodologies iteratives de desenvolupament àgil. El resultat ha estat una aplicació funcional segons els requeriments inicials.</p> <p>En definitiva, durant tot el procés d'elaboració d'aquest treball s'ha anat veient totes les fases intermèdies en l'elaboració d'un producte de programari i s'ha</p>	

experimentat com és seguir una planificació des del principi fins al final i corregint aquelles desviacions que s'han anat produint durant la seva elaboració.

Abstract (in English, 250 words or less):

More and more often, users have available several electronic devices with different services synchronized between them, through communications networks. One of these devices can access the contents of others and vice versa, since the data shared by these devices are really in the "cloud", that is, distributed on servers and accessible remotely via the Internet.

Among the different possible services, one of them could be an alarm system with extra associated capabilities. An application with these characteristics could have a universal value due to the large set of uses that could be used in many fields and contexts.

For the making of this project, several methodologies are used in the planning, including user-centered design in terms of user interaction and technical design of the application structure as well as of the database model and finally the implementation of the product using iterative methodologies for agile development. The result has been a functional application according to the initial requirements.

In short, throughout the process of doing this project, I have seen all the intermediate phases in the making of a software product, and I have experienced how it is to follow a planning from the beginning to the end and correcting the deviations that have occurred during its development.

Índex de continguts

1. Introducció.....	6
1.1. Context i justificació del Treball.....	6
1.2. Objectius del Treball.....	10
1.3. Enfocament i mètode seguit.....	12
1.4. Planificació del Treball.....	13
1.5. Breu sumari de productes obtinguts.....	14
1.6. Breu descripció dels altres capítols de la memòria.....	15
2. Disseny centrat en l'usuari (DCU).....	16
2.1. Usuaris i context d'ús.....	16
<i>Mètodes d'indagació.....</i>	<i>16</i>
<i>Fitxes dels diferents usuaris identificats.....</i>	<i>18</i>
2.2. Disseny conceptual.....	20
<i>Escenaris d'ús.....</i>	<i>20</i>
<i>Fluxos d'interacció.....</i>	<i>22</i>
2.3. Prototipatge.....	23
<i>Sketches.....</i>	<i>23</i>
<i>Prototipus horitzontal.....</i>	<i>24</i>
2.4. Avaluació.....	27
<i>Preguntes d'informació sobre l'usuari.....</i>	<i>27</i>
<i>Tasques a realitzar per l'usuari.....</i>	<i>27</i>
<i>Preguntes sobre les tasques a realitzar per l'usuari.....</i>	<i>27</i>
3. Disseny tècnic.....	29
3.1. Definició dels casos d'ús.....	29
3.2. Disseny de l'arquitectura.....	32
<i>El diagrama UML corresponent al disseny de la base de dades.....</i>	<i>32</i>
<i>El diagrama UML corresponent al disseny de les entitats i classes.....</i>	<i>32</i>
4. Implementació.....	33
4.1. Desenvolupament.....	33
<i>Metodologies utilitzades.....</i>	<i>33</i>
<i>Descripció de les eines utilitzades.....</i>	<i>33</i>
4.2. Serveis externs i bases de dades.....	34
<i>Firestore Authentication.....</i>	<i>34</i>
<i>Firestore Cloud Firestore.....</i>	<i>35</i>
<i>Firestore Cloud Storage.....</i>	<i>35</i>
4.3. Proves.....	37
4.4. Producte final.....	38
5. Conclusions.....	41
6. Índex de figures.....	42
7. Referències.....	43

1. Introducció

1.1. Context i justificació del Treball

Molt abans que hagués arribat al gran públic la tecnologia actual, com per exemple amb els mòbils o rellotges intel·ligents, els ciutadans ja tenien la possibilitat de programar dispositius com era la d'un arcaic rellotge-despertador de corda per poder-se despertar a una hora determinada.

Més tard han anat sorgint noves tecnologies que han donat una opció similar: programar el vídeo per enregistrar el nostre programa de TV favorit, posar una alarma al forn que ens avisi que ja està cuinat o posar un temporitzador perquè la calefacció s'engegui mitja hora abans d'arribar a casa, etc.

El tipus d'aplicació Android que desenvoluparia tendria una utilitat similar que permetés activar, desactivar o repetir diverses funcionalitats —com el l'alarma d'un despertador— i també fer-ho remotament gràcies a la pràcticament ubíqua disponibilitat de dispositius connectats a internet i encara més en un futur amb la *IoT*¹.

Aquesta aplicació és la que es pot englobar amb el nom de "tasques programades sincronitzades". Es vol deixar oberta la possibilitat a una futura possible extensió de les funcionalitats o tasques i per tant una atenció especial a la flexibilitat que ha de tenir el disseny perquè en un futur es puguin anar incorporant noves "tasques comunes" i no només les inicialment implementades. És important que sigui una aplicació que tingui marge de millora i creixement futur.

Tasques programades sincronitzades

El nom de "tasques programades" prové dels *Job schedulers*[1] existents ja des dels inicis dels sistemes operatius i que actualment es troben en els més populars com *Windows* o *Linux* amb els noms de *Windows Task Scheduler*[2] en del primer i *at* i *cron* en el del segon.

En el meu cas no seria per programar tasques típiques d'un administrador de sistemes sinó coses comunes que podria un usuari típic. Aquestes tasques programades típiques podrien ser comunes (fent el disseny extensible per afegir-n'hi més en un futur) com:

→ Fer saltar una alarma a una hora determinada i a partir d'ella cada X hores/minuts/segons.

★ Un exemple seria la d'un usuari té un mòbil i una tablet i vol

1 IoT (Internet of Things): Internet de les coses.

despertar-se a una hora determinada. El que fa és programar la app amb el mòbil des de la sala d'estar on el deixa a carregar-se tota la nit. Al dormitori té la tablet que ja està sincronitzada amb l'alarma i ja no haurà de configurar-la quan arribi al dormitori per dormir.

→ Fer una foto a una hora determinada i també en subsegüents intervals.

★ Un exemple seria el d'un usuari que està al carrer amb el mòbil i a casa té una tablet col·locada de tal forma que podria fer una foto de l'entrada per vigilar-ne l'accés. Vol que a partir d'un moment determinat faci fotos cada 5 minuts i les desi al núvol per poder ell visualitzar-les a qualsevol part. En qualsevol moment pot deshabilitar que segueixi fent fotos o canviar-ne els paràmetres.

El terme de "sincronitzades" és donat perquè remotament es poguessin activar, desactivar, modificar, etc. i, en l'exemple de la segona funcionalitat mencionada anteriorment, accedir a les fotos fetes.

No existeixen aplicacions a la botiga de *Google Play* que tinguin els mateixos objectius i que agrupin les mateixes funcionalitats però n'he trobat dues que incorporin característiques similars:

- En el cas de la funcionalitat de l'alarma hi ha una aplicació similar que es diu **Timely** i que és capaç de sincronitzar les alarmes en diversos dispositius.
- En el cas de fer fotos remotament he trobat casos però són molt especialitzats en seguretat o de tipus *time lapse*. Tot i així n'he trobat una amb una funcionalitat similar **Timer Camera** però sense funcionament remot.

Descartar com a altres aplicacions que poden semblar similars a al que jo proposo i però el que realment fan és executar altres aplicacions o scripts en un esdeveniment o moment donat. Alguns exemples:

- **TaskBomb task scheduler** de Ken Fehling.
- **Automate** de LlamaLab.
- **RuleBot: Automation** de KF Software House.
- **Advanced Task Manager** de INFOLIFE LLC.
- **Tasker** de joaomgcd.

Timely

Aquesta aplicació està desenvolupada per l'empresa suïssa *Bitspin GmbH*, la versió actual és la 1.3.2 actualitzada l'agost del 2018 i té més de 10 milions de descarregues.

És la típica aplicació de rellotge amb alarma, temporitzador, etc. amb diverses opcions avançades de personalització entre que la que destaca l'opció de sincronitzar les alarmes entre els diversos dispositius que tinguin instal·lada l'aplicació i registrat el mateix usuari. Com diu la pròpia descripció del producte: "*Gestiona, descarta y pausa alarmas en todos tu dispositivos a la vez. La sincronización en la nube permite una experiencia transparente entre teléfonos y tabletas. Adicionalmente, todo se respalda*"

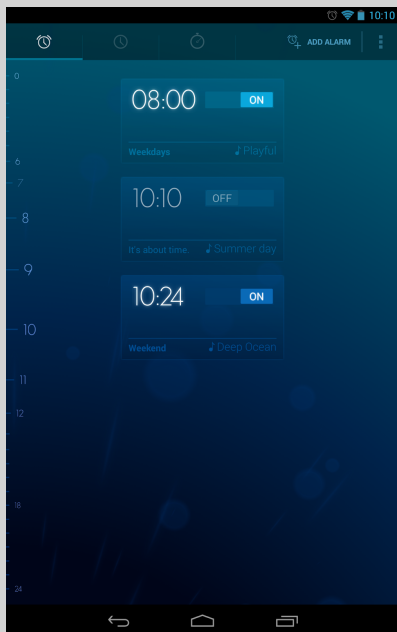


Figura 1. Llista d'alarmes de Timely.

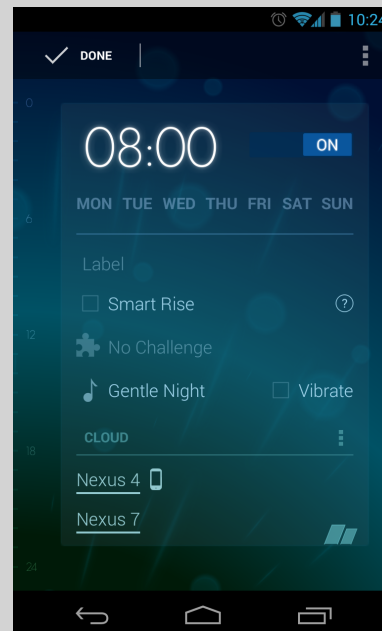


Figura 2. Configuració d'una alarma a Timely.

Timer Camera

Està desenvolupada per l'empresa canadenca *Three Starfish*, té més de 100 mil descàrregues i la última versió és la 10.10.67 actualitzada el juliol del 2018.

És una aplicació especialitzada en fotografia que, entre les diferents característiques específiques de control de la càmera, disposa d'una opció de fer fotos a hores i intervals específics. Per contra, no es pot configurar remotament.

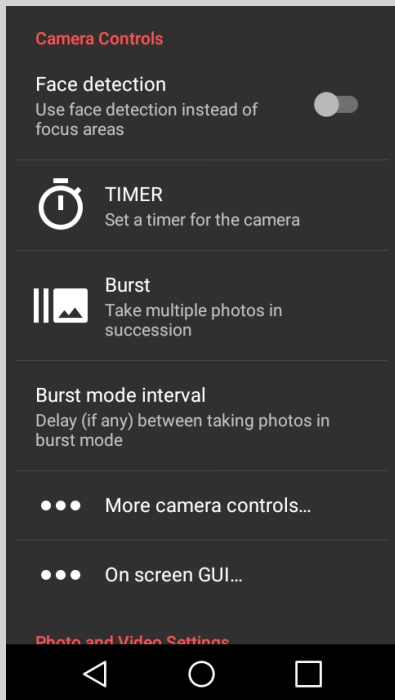


Figura 3. Configuració de Timer Camera.

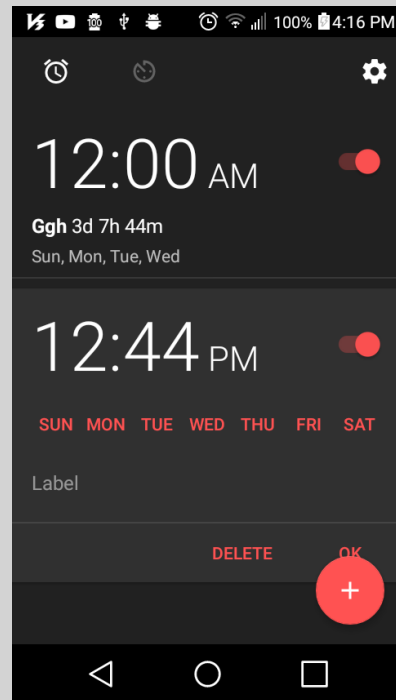


Figura 4. Llista d'alarmes a Timer Camera.

1.2. Objectius del Treball

Com a objectius principals:

- Introduir-me en el desenvolupament d'aplicacions en dispositius mòbils.
- Adquirir un coneixement suficientment ampli tant teòric com pràctic de la plataforma Android que em permeti establir uns fonaments i permetre'm, en un futur, crear aplicacions més complexes.
- Aprofundir en el coneixement del llenguatge Java.
- Aportar una nova forma d'usar i pensar les aplicacions més enllà d'utilitzar el mateix dispositiu com a introducció preliminar al que en el futur siguin tots tipus de dispositius connectats entre sí en el que l'important sigui el servei que doni l'aplicació i no tant en quin moment quin dispositiu s'estigui utilitzant.
- Conèixer característiques més tècniques dels dispositius al haver d'utilitzar la API d'Android de la càmera, so, tasques en segon pla i paral·leles per poder adaptar-les a les funcionalitats de l'aplicació.
- Conèixer i desenvolupar codi per la app utilitzant els serveis on-line que proporciona Firebase i poder aconseguir aplicacions connectades.

Com a funcionalitats principals de l'aplicació:

- L'usuari podrà configurar una alarma a una hora determinada, podrà elegir quin timbre sonarà, podrà elegir en quin dispositiu sonarà dels que que estiguin registrats al servei d'autenticació i sincronització. A partir de la hora configurada podrà seguir sonant en un interval determinat si així ho vol l'usuari. Podrà deshabilitar-ne i creant-ne de noves.
- L'usuari podrà configurar una alarma a una hora determinada perquè la càmera del dispositiu elegit, entre els disponibles, realitzi una fotografia a una hora determinada i seguint realitzant-ne en un interval determinat si així ho vol l'usuari. Podrà deshabilitar-ne i creant-ne de noves. Les imatges es podran visualitzar des de qualsevol dispositius i usuaris registrats a l'aplicació.
- La part comuna de gestió d'autenticació de l'usuari es faria amb Firebase Authentication i la sincronització de les configuracions de temps i temporització amb Cloud Firestore o Realtime Database. La part específica de la configuració amb la API d'Android de la gestió de la creació de fotografies i d'emmagatzemament d'elles a Firebase Cloud Storage.

- En principi l'aplicació s'executarà sobre dispositius Android estàndard (de tipus smartphone i tablet) però amb modificacions en la presentació es podria adaptar per a altres dispositius que utilitzen Android com a sistema operatiu com són els casos dels Smartwatches (WearOS) i els SmartTVs (AndroidTV).
- Utilitzar serveis externs de Firebase com Cloud Firestore, Cloud Storage i Authentication, i ja està suportat per *iOS*, el framework *Flutter* i *JavaScript* pel *Web*, quedaria oberta la porta a futures implementacions sobre aquestes plataformes per ampliar encara més el nombre d'usuaris potencials al que arribar.

1.3. Enfocament i mètode seguit

L'estratègia de realització de l'aplicació serà a partir d'un producte completament nou però utilitzant frameworks auxiliars o biblioteques, si s'escau, que m'ajudin a crear l'aplicació reutilitzant funcionalitats ja creades. Es tindrà en compte el tipus de llicències d'aquestes utilitats.

Per poder desenvolupar un producte ja existent hauria d'existir-ne un amb una funcionalitat similar i en qualsevol cas ser de codi obert, cosa que no he trobat.

1.4. Planificació del Treball

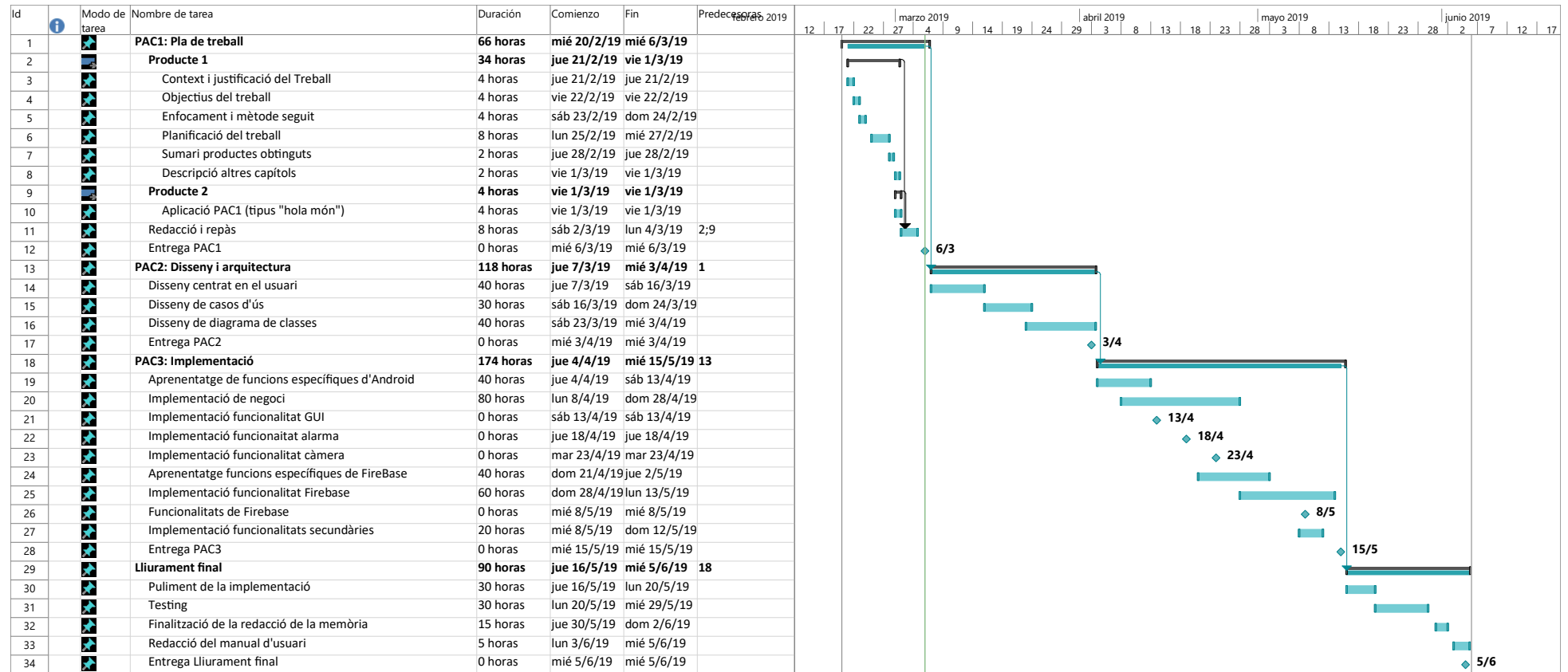


Figura 5. Planificació del treball utilitzant un diagrama de Gantt.

El calendari de dedicació creat és de 16 hores en dies de la setmana laboral i 12 hores en caps de setmana.

1.5. Breu sumari de productes obtinguts

- Aplicació Android (binari instal·lable i codi font).
- Memòria del projecte.
- Manual d'usuari i compilació.
- Presentació.

1.6. Breu descripció dels altres capítols de la memòria

En els següents capítols introduiré tant els disseny i arquitectura de l'aplicació. També faré una descripció del disseny de la usabilitat de l'aplicació.

Les decisions d'implementació i una mostra del resultat final es descriuran als últims capítols en el mateix ordre cronològic que la realització de les PACs.

2. Disseny centrat en l'usuari (DCU)

2.1. Usuaris i context d'ús

Mètodes d'indagació

D'entrada es descarten els mètodes quantitius perquè requereixen una mostra significativa d'usuaris. Entre els mètodes qualitius tenim les observacions, les entrevistes i les dinàmiques de grup. Ens decantem per la *comparativa més a fons* o *Benchmarking* ja que degut a les característiques del projecte com la mida i a la existència de productes similars és el més òptim.

L'aplicació pot ser útil per qualsevol persona independentment del gènere, edat, nacionalitat, interessos, aficions, etc. És una aplicació de funcionalitat bàsica i universal, on gairebé tothom n'ha utilitzat alguna amb un funcionament similar tant com a despertador com per a fer fotos, crec que el millor és ampliar la comparativa de les aplicacions esmentades en l'apartat 1, especialment posant l'accent en les funcionalitats en comú.

Timely

En el cas de Timely, que ens serviria de model, el primer que ens presenta quan hem acabat d'instal·lar-la és la tria d'un compte de correu electrònic on volem que se'ns sincronitzi l'aplicació (es pot configurar posteriorment quan l'aplicació ja ha estat oberta). Després ens mostra la pantalla principal que és un rellotge.

Fent *swipe* cap a l'esquerra ens mostra un llistat d'alarmes on en podem crear una de personalitzada: hora i minuts, dia de la setmana que s'apliqui l'alarma, tipus de timbre i altres opcions relacionades i finalment la funcionalitat estrella, que la fa especial i diferent a les altres, i és l'opció de triar en quin dispositiu ha de sonar.

Aquí apareix una llista amb els dispositius on està instal·lada l'aplicació registrada amb el mateix compte de correu electrònic i, en cas de que es tinguin diversos dispositius, quin d'ells vol que soni.



Figura 6. Vista dels dispositius sincronitzats a Timely.

Més enllà d'això disposa de *Snooze*, *Timer* i *Stopwatch* com les altres aplicacions del mateix tipus. Altres opcions menors seria de poder elegir el

sistema 24h o AM/PM i configuracions estètiques diverses on hi posa especial èmfasi.

Amb més de 10 milions de descàrregues pot considerar-se una de les aplicacions del mateix tipus amb més èxit. Per comparació amb l'aplicació Google Clock que té més de 100 milions de descàrregues. Potser aquesta funcionalitat que el fa diferent no és suficientment atractiu per la majoria de gent.

Timer Camera

Igualment amb el que passa amb l'aplicació d'alarma, tothom que tingui un smartphone en té una instal·lada una aplicació per fer fotos i l'ha fet servir alguna vegada. Ni ha que donen més opcions a l'hora de fer fotografies com per exemple exposició, sensibilitat, colors, etc. però normalment són per fer fotos instantàniament o amb un temporitzador d'alguns segons per allunyar-se de la càmera i fer-se un autoretrat.

El que és menys comú és poder programar una o diverses fotografies amb hores o dies d'antelació.

Aquesta aplicació permet crear dos tipus d'anticipació de la presa fotogràfica, una es de tipus alarma (hora i minuts, no dia) i l'altra és un temporitzador més flexible que el de les altres aplicacions, fins més de 100 hores. S'ha de destacar que en les alarmes que es creïn no existeix possibilitat de modificar-ne el temporitzador però si l'alarma: s'haurà d'esborrar el temporitzador i crear-ne un de nou. Una vegada s'arranca una alarma apareix una notificació a la barra superior (no se com es diu). He notat que té diversos *bugs* el tema de la temporització. No he aconseguit fer cap fotografia ni utilitzant l'alarma ni el temporitzador. Quan ho he provat s'ha penjat l'aplicació en els dos casos.

Altres funcionalitats més enllà del temporitzador són l'elecció de càmera (frontal i posterior habitualment), temps exposició sensibilitat, mode ràfega i gravació de vídeo.

Al contrari que amb el cas de *Timely*, a part del fet que no funciona al meu dispositiu la funcionalitat de temporitzador/alarma, es nota que no és un producte professional la qualitat deixa bastant que desitjar i és com a aplicació de càmera és inferior a la que duen per defecte els mòbils. Poc intuïtiva i baixa qualitat de les fotografies.

Fitxes dels diferents usuaris identificats

Usuari de la funció d'alarma sonora	
<i>Característiques del perfil</i>	Qualsevol persona de qualsevol condició que necessiti activar una alarma sonora i tingui un o més d'un dispositiu i vulgui sincronitzar-los.
<i>Contexts d'ús</i>	Pot ser utilitzat en qualsevol lloc i moment però el més habitual és una alarma sonora per fer-la servir de mode despertador.
<i>Anàlisi de tasques</i>	<ul style="list-style-type: none"> • Registrar-se. • Crear una alarma sonora elegint data, hora i dispositiu on ha d'executar-se (si n'hi ha més d'un). • Editar una alarma, modificant data, hora i dispositiu on ha d'executar-se (si n'hi ha més d'un). • Eliminar una alarma. • Donar l'opció a repetir l'alarma indefinidament en un interval elegit per l'usuari en hores, minuts i segons. • Elegir sistema 24h o AM/PM.
<i>Característiques o elements</i>	L'habitual en aquests casos: dia, hora i repetició. També l'opció d'elegir la visualització de les hores segons les condicions geogràfiques o culturals.

Usuari de la funció fotogràfica	
<i>Característiques del perfil</i>	Qualsevol persona de qualsevol condició que necessiti tenir coneixement gràfic del que passa en un indret allunyat on hi tingui un altre dispositiu.
<i>Contexts d'ús</i>	<p>Una persona que vulgui realitzar una fotografia o una seqüència de fotografies en uns temps determinats per controlar els accessos, etc. per motius de seguretat.</p> <p>O més generalment, vol realitzar fotografies a uns temps determinats, com per exemple:</p> <ul style="list-style-type: none"> • Observadors atmosfèrics. • Observadors d'evolució de la naturalesa (creixement d'una planta). • Curiosos i experimentadors.
<i>Anàlisi de tasques</i>	<ul style="list-style-type: none"> • Registrar-se. • Crear una alarma fotogràfica elegint data, hora i dispositiu on ha d'executar-se (si n'hi ha més d'un). • Editar una alarma, modificant data, hora i dispositiu on ha d'executar-se (si n'hi ha més d'un). • Eliminar una alarma. • Donar l'opció a repetir l'alarma indefinidament en un interval elegit per l'usuari en hores, minuts i segons. • Elegir sistema 24h o AM/PM.

<i>Característiques o elements</i>	Utilització de les opcions per defecte de la càmera: flaix, autofocus, etc. Silenciament del disparador. Intentar minimitzar la utilització d'energia en el dispositiu remot: apagant la pantalla, etc. Visualització de les fotografies emmagatzemades al núvol.
------------------------------------	--

2.2. Disseny conceptual

Escenaris d'ús

Escenari 1	
<i>Perfil d'usuari</i>	Usuari de la funció d'alarma sonora.
<i>Context</i>	Un usuari està a la sala d'estar al vespre i està utilitzant on té el mòbil i l'està utilitzant. També té una tablet al dormitori.
<i>Objectius</i>	Vol despertar-se demà a les 6:30 utilitzant la tablet però programant-la des del mòbil mentre està a la sala d'estar.
<i>Tasques</i>	<ol style="list-style-type: none"> 1) Obrir l'aplicació del mòbil. 2) Crear una nova alarma <u>de tipus sonora</u>. 3) Elegir demà com a dia i les 6:30 com a hora. 4) No indica que s'hagi de repetir l'alarma. 5) Elegeix que l'alarma s'activi a la tablet (per defecte és al mateix dispositiu). 6) Activa l'alarma perquè soni.
<i>Informació</i>	<ul style="list-style-type: none"> • Data i hora. • Dispositiu on ha de sonar.
<i>Funcionalitat</i>	<ul style="list-style-type: none"> • Elegir un nom per l'alarma (opcional) • Elegir el tipus d'alarma. • Seleccionador de data i hora. • Seleccionador de repetició si s'escau. • Seleccionador de dispositiu. • Habilitació/deshabilitació d'activació de l'alarma. • Creació/modificació/eliminació de l'alarma. • Sincronització automàtica de les alarmes en tots els dispositius.
<i>Desenvolupament</i>	L'usuari obri l'aplicació del mòbil i crea una nova alarma de tipus sonora sense nom. Elegeix que soni demà a les 6:30. També elegeix que l'alarma es dispari a la tauleta per això selecciona la tablet com a dispositiu. Finalment activa l'alarma.

Escenari 2	
<i>Perfil d'usuari</i>	Usuari de la funció fotogràfica.
<i>Context</i>	Un usuari està al carrer amb el mòbil i a casa té una tablet col·locada de tal forma que podria fer una foto de l'entrada per vigilar-ne l'accés.
<i>Objectius</i>	Vol que a partir de les 8 del matí, moment que se'n va a treballar, faci fotos cada 5 minuts i les desi al núvol per poder ell visualitzar-les. En qualsevol moment pot

	deshabilitar que segueixi fent fotos o canviar-ne els paràmetres.
<i>Tasques</i>	<ol style="list-style-type: none"> 1) Obrir l'aplicació del mòbil. 2) Crear una nova alarma de <u>tipus fotogràfica</u>. 3) Elegir demà com a dia i les 8:30 com a hora. 4) Elegir un interval de 5 minuts de repetició. 5) Elegeix que l'alarma s'activi a la tablet (per defecte és al mateix dispositiu). 6) Activa l'alarma perquè faci les fotos quan s'executi.
<i>Informació</i>	<ul style="list-style-type: none"> • Data, hora i interval de repetició. • Dispositiu amb el qual s'han de fer les fotos.
<i>Funcionalitat</i>	<ul style="list-style-type: none"> • Elegir un nom per l'alarma (opcional). • Seleccionador de data i hora. • Seleccionador de repetició si s'escau. • Seleccionador de dispositiu. • Habilitació/deshabilitació d'activació de l'alarma. • Creació/modificació/eliminació de l'alarma. • Sincronització automàtica de les alarmes en tots els dispositius.
<i>Desenvolupament</i>	L'usuari obri l'aplicació del mòbil i crea una nova alarma de tipus fotogràfica amb el nom "Vigila". Elegeix que comenci a fer fotos a partir de demà a les 8:00 i que repeteixi cada 5 minuts. També elegeix que l'alarma es dispari a la tauleta per això selecciona la tablet com a dispositiu. Finalment activa l'alarma.

Fluxos d'interacció

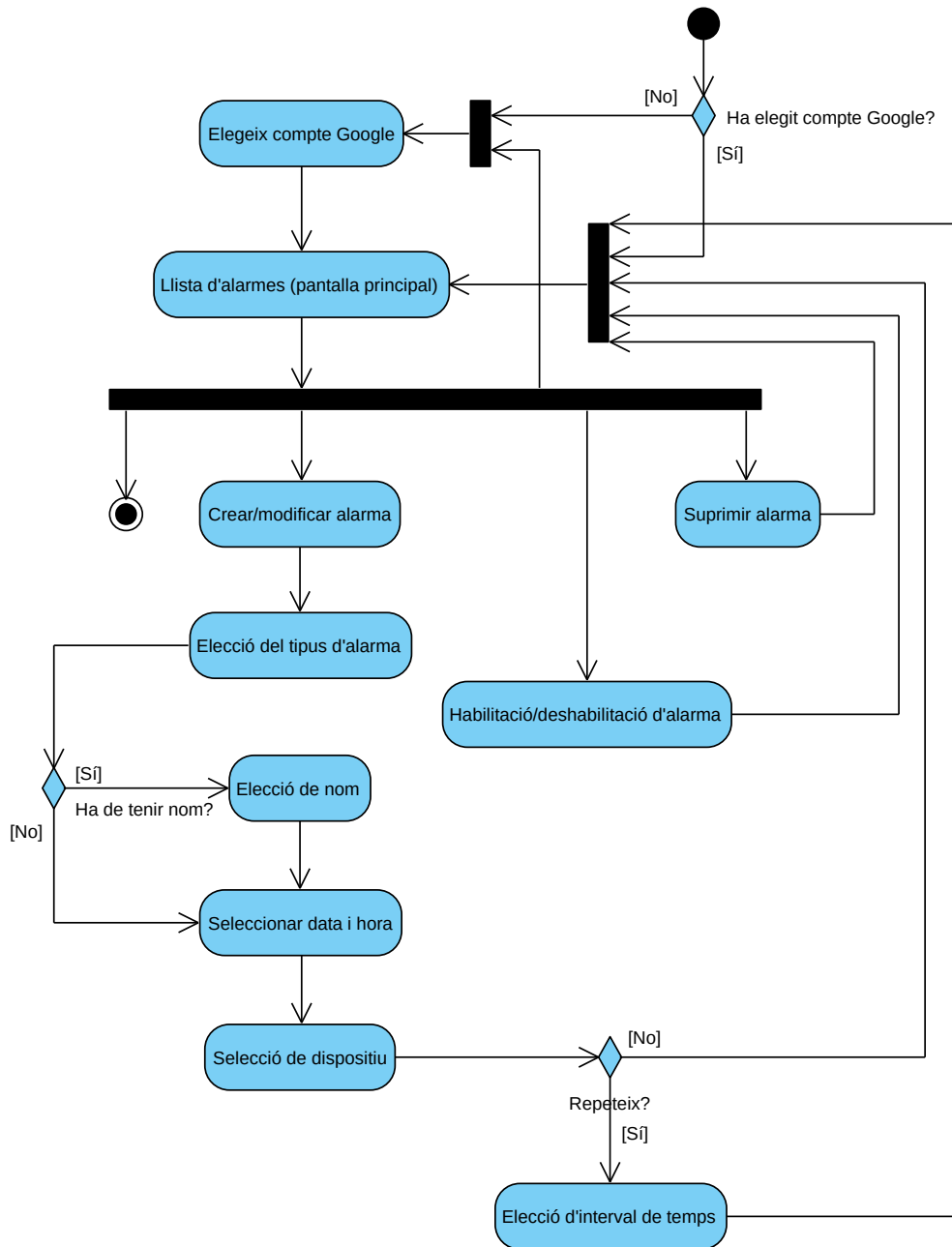
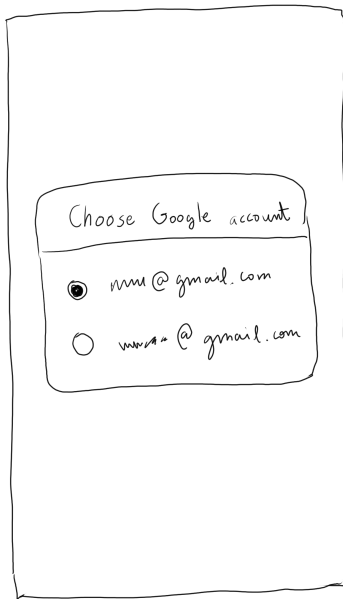


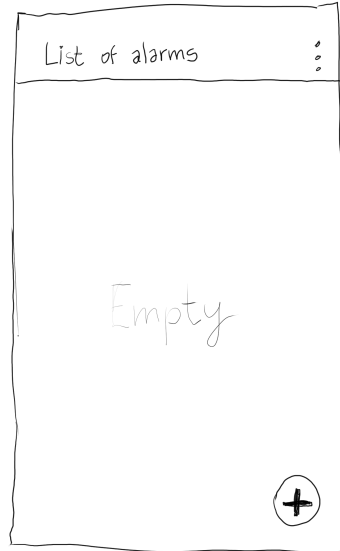
Figura 7. Flux d'interacció de la interfície d'usuari.

2.3. Prototipatge

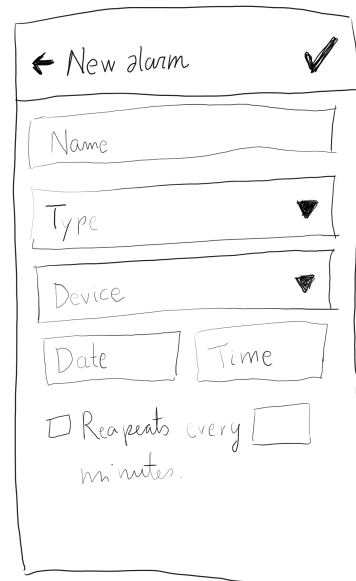
Sketches



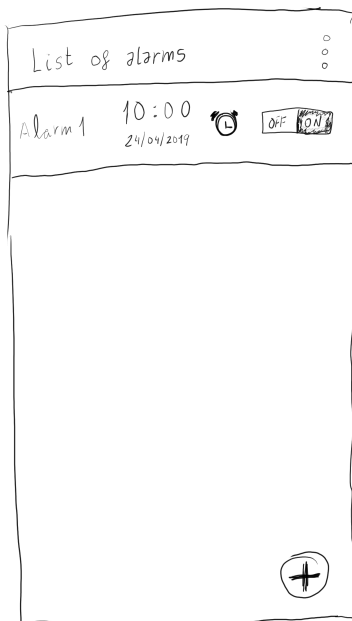
Choosing the Google account that we want to use as registering point.



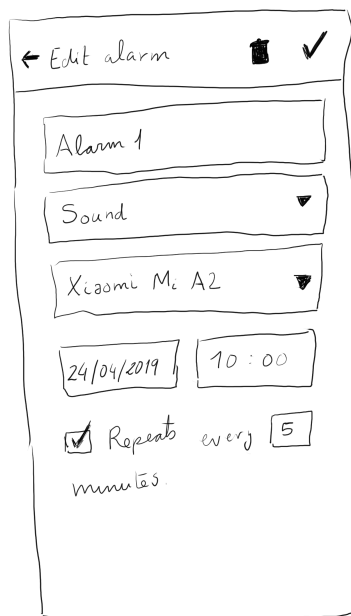
Empty alarm list



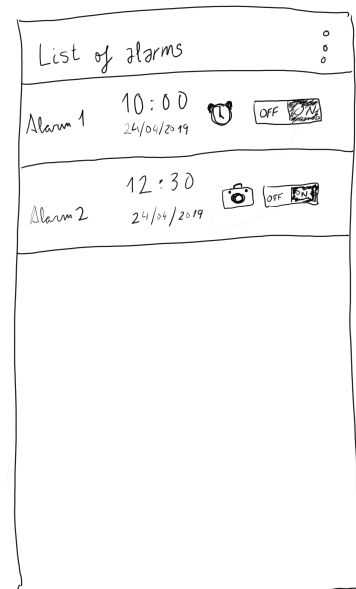
Creating a new alarm



One alarm in alarm list



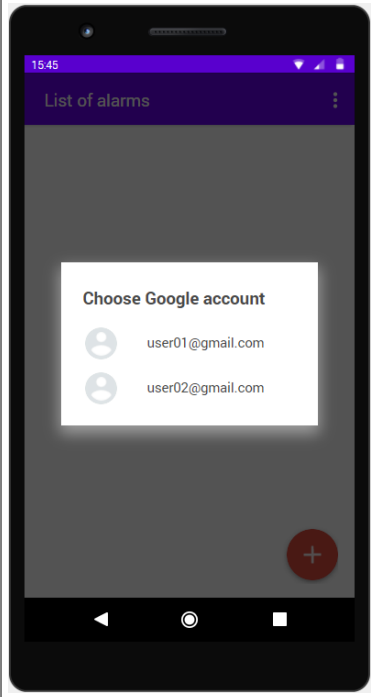
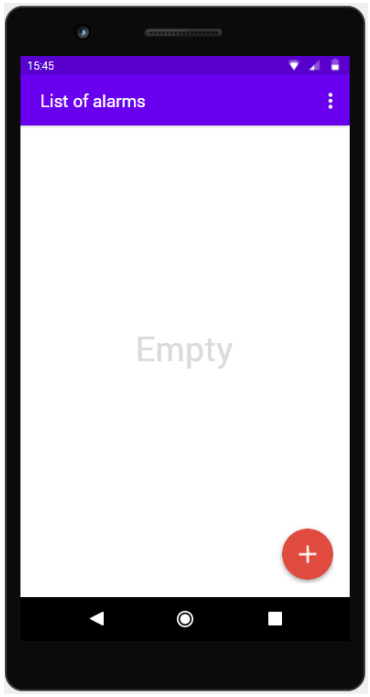
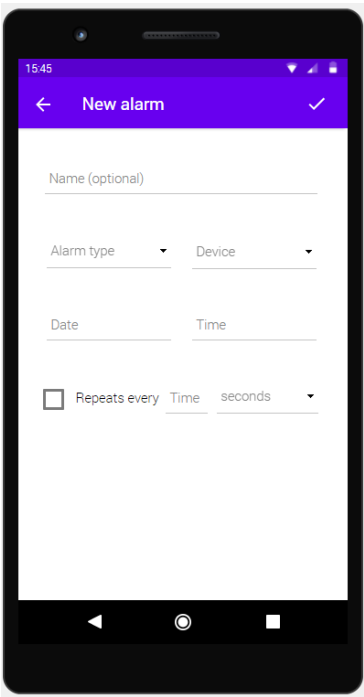

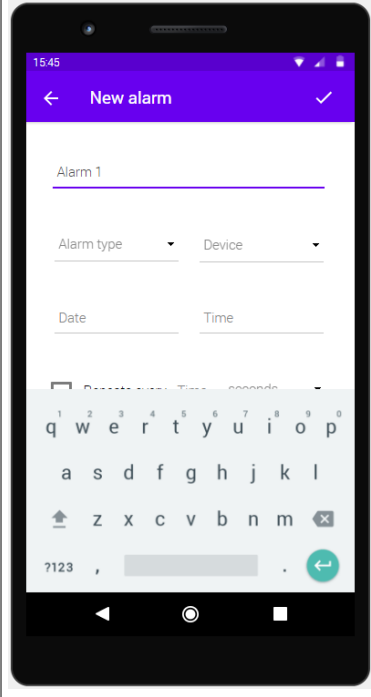
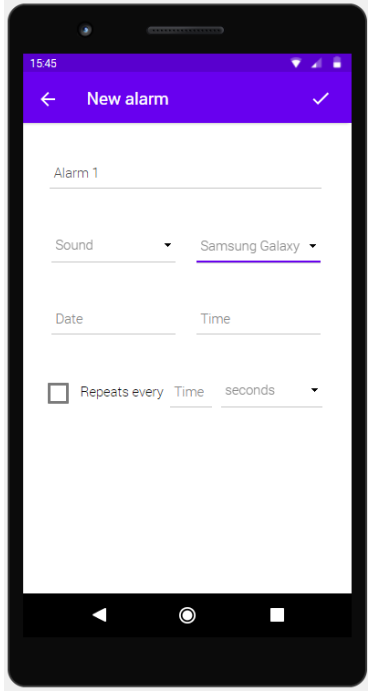
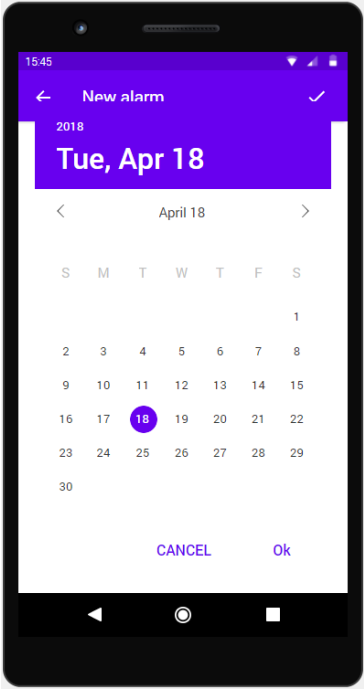
Editing existing alarm

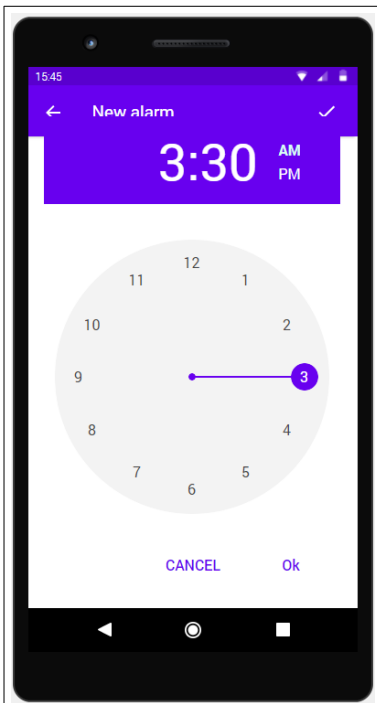
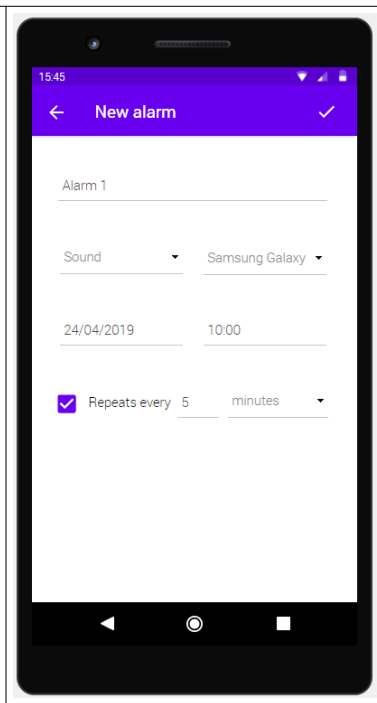
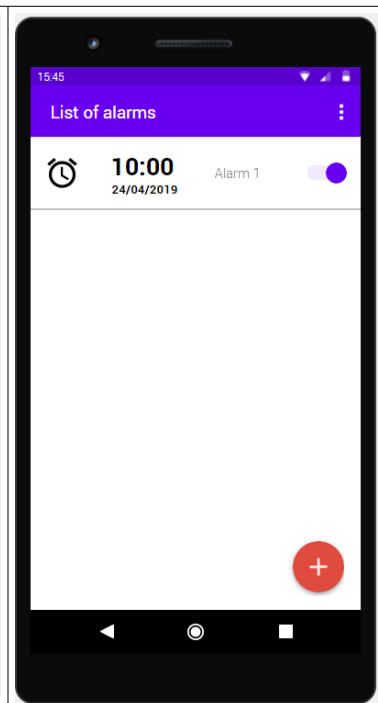


Two alarms of different kind Scheduled and active.

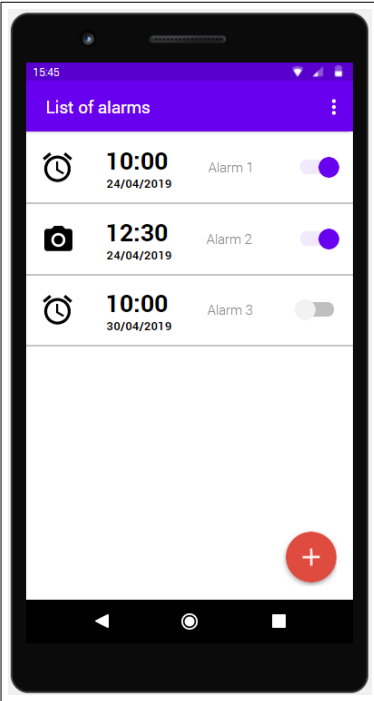
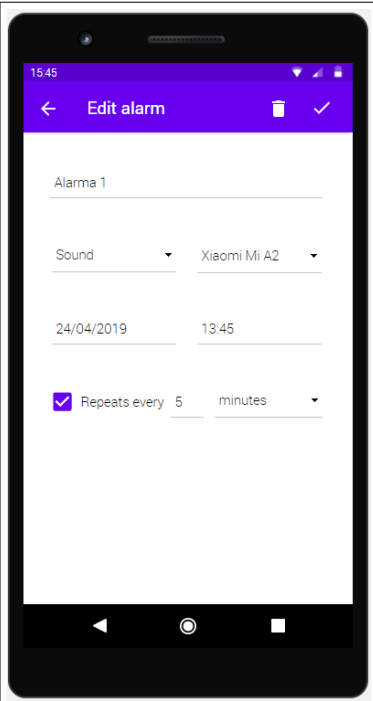
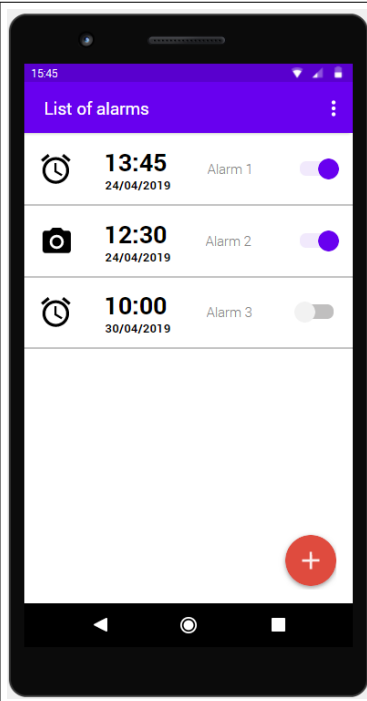
Prototipus horitzontal

Creació d'una alarma:

		
<p>A l'inici, després d'haver instal·lat l'aplicació apareix l'elecció de compte Google. També es pot canviar en qualsevol moment des de la icona de la  barra d'activitat.</p>	<p>A l'inici ens trobem amb la llista d'alarmes buida i un botó per crear-ne de noves.</p>	<p>En aquesta pantalla de creació d'alarma tipus formulari introduïrem el nom, tipus, dispositiu on volem que s'executi, data, hora i si volem que es repeteixi i cada quan.</p>
		
<p>Quan volem introduir el nom i activem el camp ens apareix automàticament el teclat virtual.</p>	<p>L'elecció de tipus d'alarma i de dispositiu es realitza mitjançant uns menús tipus <i>drop-down</i>. En aquest cas elegim de tipus <i>Sound</i> i un mòbil <i>Samsung Galaxy</i>.</p>	<p>Per la data ens apareix un <i>widget</i> propi d'Android.</p>

		
<p>Per l'hora ens apareix un <i>widget</i> propi d'Android.</p>	<p>Per acabar de configurar l'alarma elegim si volem que es repeteixi i la freqüència d'aquesta en segons, minuts o hores. Per cancel·lar-la i tornar enrere espitgem sobre ← per crear-la sobre ✓.</p>	<p>Finalment només ens quedarà activar-la movent el botó horitzontal de la part dreta.</p>

Edició d'una alarma:

		
<p>Per editar una alarma, tocarem la part de l'esquerra de l'alarma de la llista d'alarmes i sens obrirà una altra vegada el formulari d'edició.</p>	<p>Aquí editem l'alarma com desitgem i si volem cancel·lar l'edició i tornar enrere espitgem sobre ←, per esborrar l'alarma sobre 🗑️ i per desar el canvi sobre ✓.</p>	<p>Finalment quan tornem a la llista d'alarmes ens apareixen totes les alarmes que tenim associades al compte de Google. Tant les que s'executaran localment com remotament.</p>

2.4. Avaluació

Per avaluar la usabilitat de l'aplicació s'avaluaran tres apartats que són: preguntes referents al propi usuari, tasques que l'usuari ha de realitzar i preguntes sobre aquestes tasques.

Preguntes d'informació sobre l'usuari

De cada usuari ens interessen un conjunt de característiques personals lligades al seu potencial ús de l'aplicació:

- Edat, gènere i altres trets personals i culturals.
- Grau d'experiència manejant dispositius electrònics.
- Quines capacitats desitjaria en una aplicació mòbil que encara no ha trobat.
- Aficions que li agradaria realitzar juntament amb les capacitats extra que li aporta els dispositius mòbils.
- Quantitat de dispositius mòbils que té i quin sistema operatiu disposen (Android i iOS principalment).

Tasques a realitzar per l'usuari

- Registrar-se a l'aplicació amb dos dispositius diferents.
- Creació d'alarmes de tots dos tipus disponibles i als dos dispositius diferents.
 - Escoltar l'alarma sonora creada en un dispositiu a l'altre.
 - Visualització de les fotografies realitzades per un dispositiu des de l'altre.
- Modificació i eliminació de les alarmes anteriorment creades.
- Una vegada registrat amb un compte, fer-ho amb un altre.

Preguntes sobre les tasques a realitzar per l'usuari

- Preguntes sobre el nivell de dificultat per portar a terme les tasques.
- Preguntes sobre la interfície gràfica i la seva riquesa visual.
- Preguntes sobre la navegabilitat entre pantalles.

- Preguntes sobre les possibles frustracions que li provoqui alguna de les tasques realitzades.

3. Disseny tècnic

3.1. Definició dels casos d'ús

Identificador	CU-001
Nom	Registrar el compte i dispositiu a l'aplicació.
Prioritat	Alta.
Descripció	Acabada d'instal·lar l'aplicació al obrir-se s'ha de vincular l'aplicació amb un compte Google per poder després sincronitzar les alarmes amb tots els dispositius del mateix usuari.
Actors	Usuari.
Pre-condicions	L'aplicació ha d'estar instal·lada i l'usuari ha de tenir un compte Google.
Iniciat per	L'usuari.
Flux	<ol style="list-style-type: none">1) L'usuari instal·la l'aplicació.2) L'usuari executa l'aplicació.3) Apareix una llista de comptes Google de l'usuari i n'ha d'escollir una.
Post-condicions	L'usuari i el dispositiu queden registrats al sistema.
Notes	<ul style="list-style-type: none">• Una vegada registrada l'aplicació amb el compte, es pot vincular a un altre compte si l'usuari així ho desitja.• Aquest procés d'instal·lació i registre de l'aplicació s'ha de seguir per cada un dels dispositius que l'usuari vulgui compartir les alarmes.

Identificador	CU-002
Nom	Donar d'alta una alarma.
Prioritat	Normal.
Descripció	Crea una nova alarma a l'aplicació del dispositiu i a la base de dades i per tant també a la resta de dispositius de l'usuari registrat.
Actors	Usuari.
Pre-condicions	L'usuari ha d'estar registrat (CU-001).
Iniciat per	L'usuari.
Flux	<ol style="list-style-type: none">1) L'usuari toca el botó d'afegir una nova alarma.2) Apareix una nova pantalla amb el formulari de creació d'alarmes amb les opcions corresponents.3) Toca el botó de desmarcar l'alarma creada.

Post-condicions	L'alarma queda enregistrada a o als dispositius sincronitzats.
Notes	<ul style="list-style-type: none"> • Si l'usuari toca l'opció de tornar enrere, el procés de creació de l'alarma s'interromp. • Tots els camps del formulari són obligatoris excepte el del nom de l'alarma i el de l'interval de repetició. El sistema no permetrà desar l'alarma si no tots els camps obligatoris estan emplenats. • No existeixen límit de creació d'alarmes. Es creen tantes alarmes noves com l'usuari desitgi. Cada nova alarma té un identificador únic.

Identificador	CU-003
Nom	Editar o esborrar una alarma ja existent.
Prioritat	Normal.
Descripció	Permet editar o esborrar una alarma ja existent i modificar-ne els seus camps.
Actors	Usuari.
Pre-condicions	Ha de ser una alarma que ja existeixi.
Iniciat per	L'usuari.
Flux	<ol style="list-style-type: none"> 1) L'usuari toca la part esquerra de l'alarma que vol modificar a la llista d'alarmes. 2) Apareix una nova pantalla amb el formulari d'edició d'alarmes amb les opcions corresponents. 3) Toca el botó de desar l'alarma modificada per acceptar els canvis.
Post-condicions	La modificació de l'alarma queda enregistrada als dispositius sincronitzats.
Notes	<ul style="list-style-type: none"> • Si l'usuari toca l'opció de tornar enrere, el procés de modificació de l'alarma s'interromp. • Si l'usuari toca l'opció d'esborrar, l'alarma s'elimina. • Tots els camps del formulari són obligatoris excepte el del nom de l'alarma i el de l'interval de repetició. El sistema no permetrà desar l'alarma si no tots els camps obligatoris estan emplenats.

Identificador	CU-004
Nom	Elecció de repetició de l'alarma i interval.
Prioritat	Baixa.
Descripció	Tant en la creació com en la modificació de les alarmes hi ha la possibilitat de fer que aquesta es repeteixi indefinidament en un interval de temps fix.

Actors	Usuari.
Pre-condicions	Estar en el procés de crear o editar una alarma (CU-002 i CU-003).
Iniciat per	L'usuari.
Flux	<ol style="list-style-type: none"> 1) A l'apartat corresponent s'activa la casella d'activació. 2) S'indica la quantitat i el tipus (segons / minuts / hores / dies).
Post-condicions	El procés de creació o modificació arriba al final amb èxit.
Notes	En cas de desactivar-se la casella, els valors han de seguir emmagatzemats per si l'usuari eventualment vol tornar-la a activar i no haver de reintroduir.

3.2. Disseny de l'arquitectura

El diagrama UML corresponent al disseny de la base de dades

Al fer-se servir *Firebase*, que és una base de dades de tipus NoSQL documental enlloc de les típiques relacionals, s'aplicarà el mètode de disseny explicat a l'article *NoSQL Database Design Using UML Conceptual Data Model Based on Peter Chen's Framework*[3]:

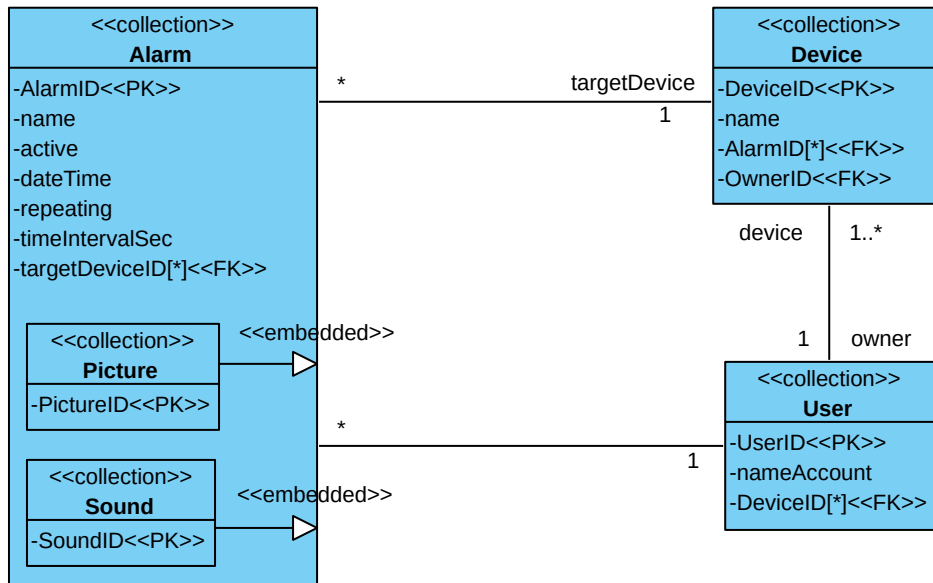


Figura 8. Model de dades de l'aplicació per una base de dades NoSQL de tipus documental.

El diagrama UML corresponent al disseny de les entitats i classes

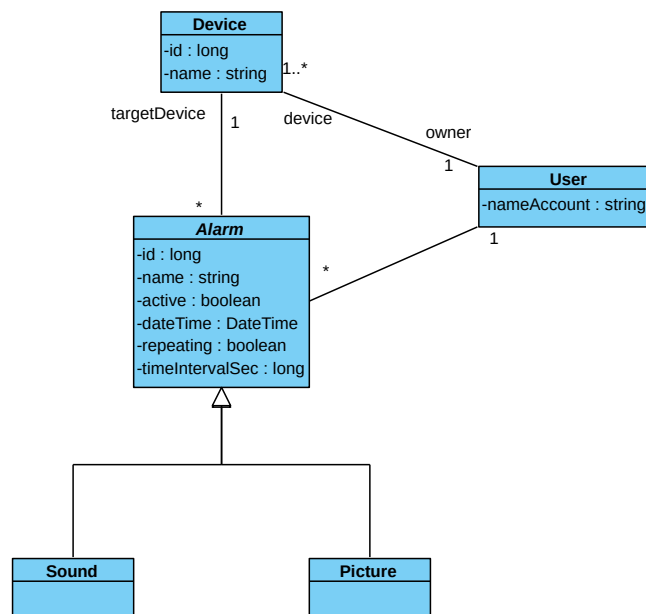


Figura 9. Diagrama de classes des del punt de vista de la informació.

4. Implementació

4.1. Desenvolupament

Metodologies utilitzades

Tot i que el desenvolupament no ha estat en grup on sol ser més habitual la seva utilització, a l'hora d'implementar l'aplicació s'ha enfocat mitjançant l'ús de mètodes de desenvolupament iteratiu i incremental.

S'ha elegit anar implementant funcionalitats en cada iteració segons la importància que tingui en el global de l'aplicació i si són funcionalitats nuclears amb les quals altres hi tenen una dependència.

També s'ha donat prioritat a que en cada iteració funcioni tota la funcionalitat establerta per aquella iteració en concret.

Les iteracions, en el procés Scrum, són els sprints i en aquest cas s'han organitzat en un total de quatre de la següent forma:

- Sprint 1: del 04/04/2019 al 21/4/2019
 - S'implementa la funcionalitat bàsica de la interfície d'usuari, part de l'accés a dades i autenticació.
- Sprint 2: del 22/04/2019 al 5/5/2019
 - Es completa i estén la implementació iniciada a l'sprint 1.
- Sprint 3 (lliurament PAC3): del 6/5/2019 al 15/5/2019
 - S'implementa el servei d'alarma sonora.
- Sprint 4 (lliurament final): del 16/5/2019 al 5/6/2019
 - S'implementa el servei d'alarma fotogràfica, s'actualitza l'estructura i es fan els últims ajustos.

Descripció de les eines utilitzades

S'ha utilitzat l'IDE AndroidStudio 3.4 en totes les fases: tant per editar el codi (en Java), com per compilar, depurar i executar l'aplicació en l'emulador.

Com a eina extra, tot i no ser imprescindible al no ser un projecte en grup, s'ha utilitzat Git[4] per crear punts de restauració en cas que hi hagués hagut alguna disminució en la qualitat del codi.

4.2. Serveis externs i bases de dades

L'aplicació fa servir diversos mòduls externs que ajuden a desenvolupar l'aplicació més ràpidament. Les biblioteques externes utilitzades s'especifiquen a l'arxiu *build.gradle* del projecte i són:

- **Firestore:** s'utilitzen els serveis de les biblioteques d'autenticació Firebase Authentication, de la base de dades Firestore, emmagatzematge al núvol Cloud Storage. Així mateix també es fa servir FirebaseUI per la implementació gràfica del registre i accés i a la visualització de les fotografies.
 - S'ha elegit pel conglomerat de serveis que proporciona i perquè permet fer actualitzacions de la base de dades en temps real i poder sincronitzar els canvis en tots els dispositius connectats immediatament, clau en aquesta aplicació.
 - La utilització d'aquests serveis té diversos trams de preus, sent la més bàsica i limitada, gratuïta.
- **JUnit:**[5] biblioteques per fer tant les proves unitàries aïllades com les instrumentades. Sota llicència pública Eclipse.
- **ThreeTenABP:**[6] llibreria que permet utilitzar les classes contingudes al paquet `java.time` de Java 8, en aplicacions que utilitzen Java 6 i 7. Són útils per tractar i transformar dates. Sota la llicència Apache 2.0.
- **EZCam:**[7] és un *wrapper* per facilitar l'ús del paquet `android.hardware.camera2` d'Android per a usos bàsics. Repositori públic però sense esmentar cap llicència de cap tipus.
- **Glide:**[8] és una llibreria que s'encerrega de gestionar la càrrega d'imatges i el procés de caching i que està incorporada al mòdul `storage` de FirebaseUI indirectament per descarregar les fotografies al sistema d'emmagatzematge al núvol de Firebase Cloud Storage.

A continuació s'aprofunditza en la utilització de Firebase ja que és cabdal la seva importància en l'aplicació:

Firestore Authentication

Proporciona serveis de servidor per autenticar als usuaris de l'aplicació. Disposa de diversos mètodes d'accés: correu+contrasenya, números de telèfon o proveïdors d'identitat com Google, Facebook, Github, etc.

Mitjançant la implementació Auth de FirebaseUI de la biblioteca, es pot integrar l'accés directament a la pròpia aplicació incorporant l'activity d'accés i no haver d'implementar-ne una de pròpia.

En l'aplicació s'ha incorporat els mètodes d'accés per email+contrasenya i amb compte Google.

Firestore Cloud Firebase

És la successora de la base de dades de Firebase Realtime Database. Comparteix amb aquesta primera les capacitats de sincronització de les dades en temps real i que utilitza un tipus de model de dades NoSQL documental però incorpora novetats com la major escalabilitat, rapidesa i facilitat d'implementació.

Hi ha tres tipus d'estructures: documents, camps i col·leccions. Cada document pot contenir camps i col·leccions que al mateix temps pot contenir altres documents, de forma jeràrquica.

Per contra, la política de preus és més restrictiva en el cas de Firestore al carregar per operacions realitzades i l'emmagatzematge a més de l'ample de banda utilitzat, que és l'únic criteri utilitzat per la utilització de Firebase Realtime Database.

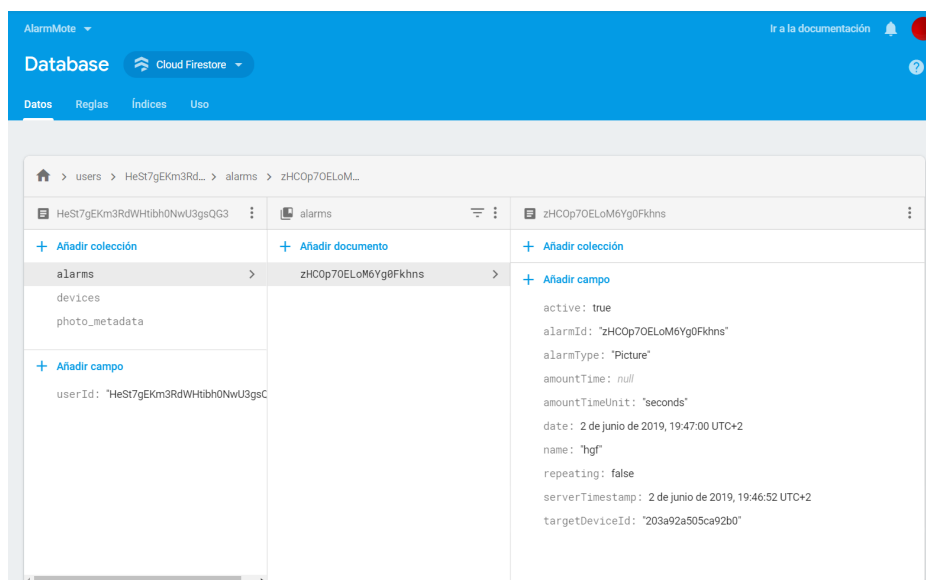


Figura 10. Consola d'administració de Firestore Firebase.

Storage Cloud Firebase

Cloud storage és un servei que s'encarrega d'emmagatzemar els fitxers dels usuaris i, en el cas d'aquesta aplicació les fotografies realitzades.

S'emmagatzemen en un dipòsit Google Cloud Storage i a més, gràcies al component storage de FirebaseUI, es facilita la descàrrega i visualització de les imatges allotjades a Cloud Storage utilitzant internament altres biblioteques, com Glide, que ja incorpora tota la funcionalitat de gestió de descàrrega per xarxa.

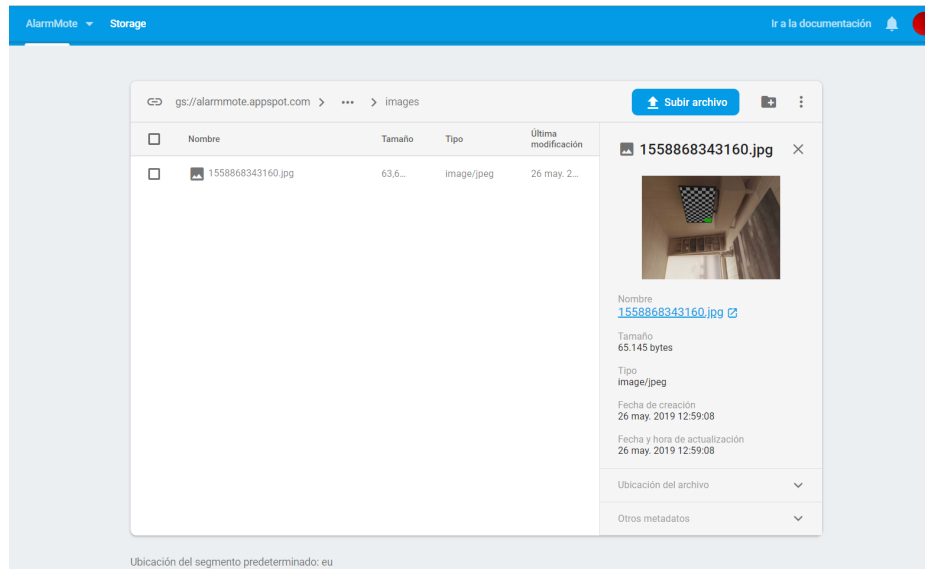


Figura 11. Consola d'administració de Firebase Cloud Storage.

4.3. Proves

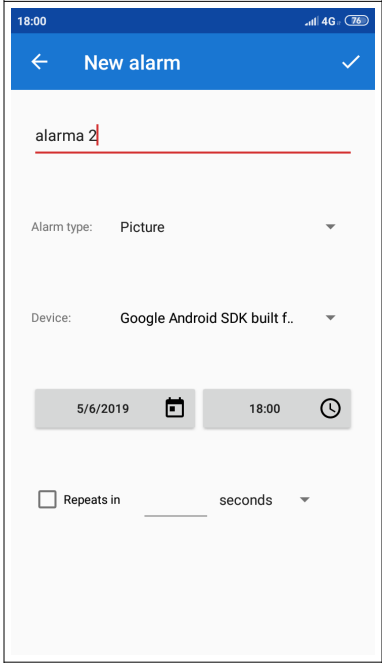
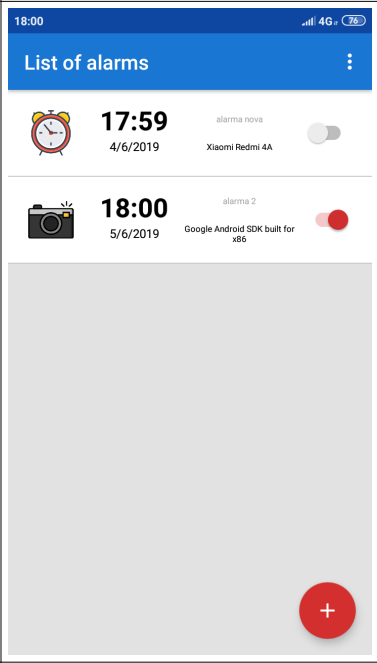
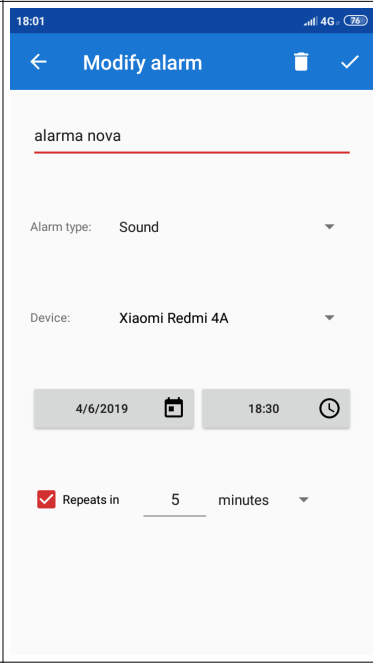
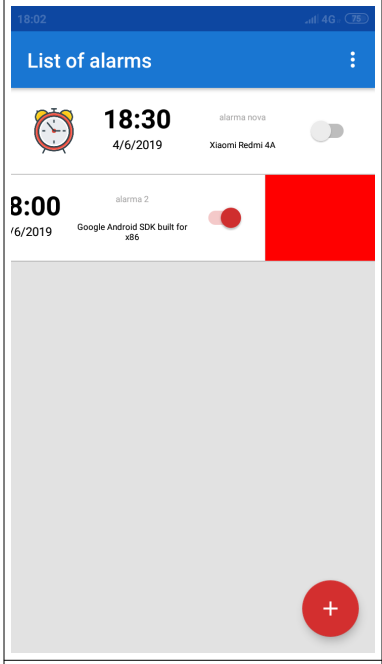
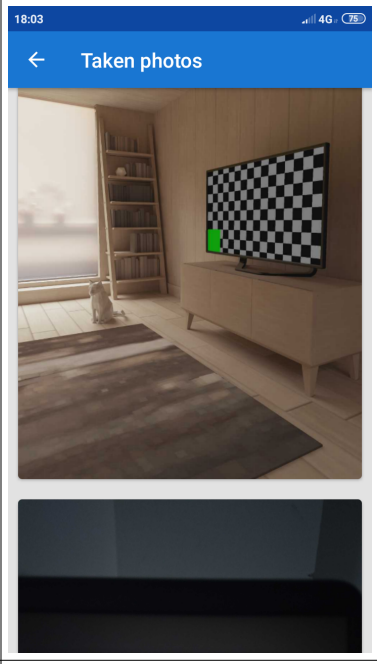
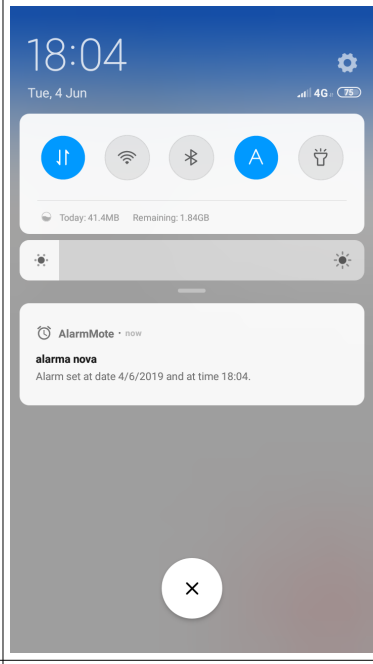
Estan implementades proves unitàries tant les aïllades de l'entorn Android com les que s'executen en l'entorn (instrumentades).

En concret s'han provat mètodes de les classes `DateTimeConversions.java` i `Alarm.java`.

4.4. Producte final

Com queda el producte final i en comparació amb els prototips creats en l'apartat de disseny centrat en l'usuari.

<p>Una vegada instal·lada l'aplicació se'ns apareix la pantalla d'entrada on hem d'elegir com volem accedir-hi. Està implantat l'accés tant amb compte de Google com amb email més contrasenya.</p>	<p>En cas de voler accedir amb compte de Google se'ns presenta una pantalla amb quina d'elles volem entrar.</p>	<p>La primera vegada que entrem ens sortirà la llista d'alarmes buida. Per crear-ne una haurem de tocar el botó +.</p>

<p>Al tocar el botó + obtindrem un formulari on establir les condicions que volem que tingui l'alarma mitjançant els seus camps. En aquest cas creem una alarma sonora.</p>	<p>Exemple d'elecció de data.</p>	<p>Exemple d'elecció d'hora.</p>
		
<p>Al crear una alarma de tipus fotogràfica elegim el tipus "Picture".</p>	<p>Les alarmes creades apareixen a la llista d'alarmes.</p>	<p>En cas de voler modificar una alarma, toquem sobre ella en la llista i se'ns apareixerà un formulari de modificació. En aquest cas volem modificar l'hora i volem que es repeteixi l'alarma en cinc minuts. També ens dona l'opció d'esborrar l'alarma.</p>
		
<p>Podem esborrar directament una alarma a la llista d'alarmes lliscant l'alarma cap a l'esquerra.</p>	<p>Les fotografies fetes es poden veure accedint-hi des del menú ☰ a l'opció "Taken photos".</p>	<p>Notificació informativa quan es dispara una alarma.</p>

5. Conclusions

Vaig començar aquest treball sense cap experiència en desenvolupar aplicacions per a Android i durant el transcurs d'aquest he anat adquirint coneixements i adonar-me'n de les grans possibilitats que té però també del grau d'experiència necessària per crear aplicacions més complexes.

Crec que he assolit els objectius plantejats inicialment al haver creat el producte final a temps amb les funcionalitats requerides. Tot i així m'agradaria haver explorat més les possibilitat d'aplicar patrons tant d'arquitectura com de disseny per fer l'aplicació més modular i extensible en un futur.

S'ha de remarcar que la dependència que té l'aplicació del conglomerat de serveis de Firebase la fa molt vulnerable a canvis que es produeixin en ells en un futur per part de Google, tant tecnològics i especialment econòmics. Al ser de pagament, a partir d'un cert nombre d'usuaris i de tràfic de dades, perquè fos viable econòmicament l'aplicació s'hauria de monetitzar d'alguna manera.

Una altra consideració és la de la preservació de la privacitat en el sistema de Firebase. La informació no està encriptada i l'administrador té accés a tota la informació que els usuaris van generant. Una via que s'hauria d'investigar seria la possibilitat d'utilitzar serveis només accessibles per l'usuari, com l'emmagatzematge d'imatges directament a Google Drive enlloc de Firebase cloud storage.

6. Índex de figures

Figura 1. Llista d'alarmes de Timely.....	8
Figura 2. Configuració d'una alarma a Timely.....	8
Figura 3. Configuració de Timer Camera.....	9
Figura 4. Llista d'alarmes a Timer Camera.....	9
Figura 5. Planificació del treball utilitzant un diagrama de Gantt.....	13
Figura 6. Vista dels dispositius sincronitzats a Timely.....	16
Figura 7. Flux d'interacció de la interfície d'usuari.....	22
Figura 8. Model de dades de l'aplicació per una base de dades NoSQL de tipus documental.....	32
Figura 9. Diagrama de classes des del punt de vista de la informació.....	32
Figura 10. Consola d'administració de Firebase Firestore.....	35
Figura 11. Consola d'administració de Firebase Cloud Storage.....	36

7. Referències

- 1: Diversos autors, Job scheduler, 2019, https://en.wikipedia.org/wiki/Job_scheduler
- 2: Microsoft, Task Scheduler, 2018, <https://docs.microsoft.com/en-us/windows/desktop/taskschd/task-scheduler-start-page>
- 3: Kwangchul Shin, Chulhyun Hwang, Hoekyung Jung, NoSQL Database Design Using UML Conceptual Data Model Based on Peter Chen's Framework, 2016, https://www.ripublication.com/ijaer17/ijaerv12n5_12.pdf
- 4: Linus Torvalds et alii, Git. Sistema de control de versions., 2019, <https://git-scm.com>
- 5: Kent Beck, Erich Gamma et alii, JUnit, 2019, <https://junit.org/junit5/>
- 6: Jake Wharton, ThreeTenABP. An adaptation of the JSR-310 backport for Android., 2019, <https://github.com/JakeWharton/ThreeTenABP>
- 7: Omar Aflak, EZCam, 2019, <https://github.com/OmarAflak/Android-Camera2-Library>
- 8: Bumptech, Glide, 2019, <https://bumptech.github.io/glide/>