

Aplicació de control d'accessos

Manuel Reche Giráldez

Resum—Aquest treball es basa en el desenvolupament d'una aplicació que permeti controlar l'accés de persones a espais restringits a través de la lectura d'un codi QR. A la vegada de controlar l'accés als espais restringits també es mantindrà un històric dels accessos fets pels usuaris a aquests espais. A més, l'aplicació contindrà un espai pels administradors, des d'on podran gestionar els usuaris, les sales i accedir als històrics d'accessos.

Paraules clau— QR(Quick Response), aplicació, espais restringits, usuaris, històric.

Resumen—Este trabajo se basa en el desarrollo de una aplicación que permita controlar el acceso de personas a espacios restringidos a través de la lectura de un código QR. A su vez, irá manteniendo un histórico de los accesos hechos por los usuarios a estos espacios. Además, la aplicación constará con un apartado para los administradores, desde donde podrán gestionar a los usuarios, las salas y acceder al histórico de accesos.

Palabras clave— QR(Quick Response), aplicación, espacios restringidos, usuarios, histórico.

Abstract—The objective of this work is the development of an application that allow the users acces control in a private areas by the scanning a QR code. At the same time while is doing the control acces the application will generate a historical file of the users acces. Moreover, this application will contain an area for the administrator of the system where they can manage the user's and the room's information and even get access to the historical access.

Index Terms— QR(Quick Response), application, private areas, users, historical.



1 INTRODUCCIÓ

AQUEST treball té com a objectiu el desenvolupament d'una aplicació que garanteixi un control d'accessos a arees privades o restringides d'espais al personal adient, com podria ser una empresa on hi ha diversos mòduls de treball i només poden accedir a ells els usuaris que tenen l'autorització per poder entrar. D'aquesta manera es podrà generar un històric amb informació dels accessos duts a terme.

A continuació, en aquest document es recolliran tots els requisits per a poder desenvolupar l'aplicació a través d'un estudi de mercat analitzant les aplicacions existents. També podrem trobar com s'ha dut a terme el desenvolupament del projecte i finalment un apartat on s'extrauran les conclusions del projecte.

- E-mail de contacte: manuel.reche@e-campus.uab.cat
- Menció realitzada: Enginyeria del Software
- Treball tutoritzat per: Marc Talló Sendra (Ciències de la Computació)
- Curs 2017/18

2 CAPTURA DE REQUISITS

2.1 Estat de l'art

Pel correcte desenvolupament del projecte es realitza un estudi de mercat per avaluar aplicacions existents i d'aquesta manera poder extreure les funcionalitats necessàries. A continuació es mostrarà la anàlisi de 3 aplicacions on podrem veure fortalezes i debilitats de la mateixa.

- **Ticketea Checkpoint:** Aquesta es una aplicació gratuïta per a la validació d'entrades on has d'introduir el codi de l'esdeveniment al que vols assistir, s'obre una llista amb tots el participants i o bé pots buscar el nom de l'assistent o pots fer un escaneig de QR per a validar la seva assistència.
 - Fortalezes: L'escaneig de QR agilitza molt el procés de validar l'assistent i per tant agilitza les cues que es poden formar a les entrades dels esdeveniments.
 - Desavantatges: No hi ha un apartat per als usuaris, és a dir, un usuari no podria tenir un llistat

d'esdeveniments pels que té entrades, esta pensada només pels organitzadors de l'esdeveniment.

- **Control de Accesos Atrápalo** Aquesta aplicació de control d'accessos és de la famosa pàgina web "Atrápalo" que ofereixen diferents serveis com entrades de cinema, restaurants, vols, etc. Aquesta aplicació et permet ficar una ID del proveïdor de servei juntament amb el teu usuari per validar la teva assistència, també permeten validar la reserva a través d'un codi QR que generen al fer la reserva amb ells.
 - Fortaleses: Igual que l'anterior, agilitzen molt el procés utilitzant l'escaneig de QR i la seva interfície és molt user friendly.
 - Desavantatges: Al ser una aplicació d'una pàgina web en concret només et podrà ser útil per les ofertes d'aquella pàgina.
- **Enterticket:** Aquesta aplicació et permet crear i gestionar els teus propis esdeveniments de forma gratuïta. La aplicació et permet fer un control d'accessos ja que te permet verificar la validesa de les entrades associades al esdeveniment a través d'una lectura ràpida de QR.
 - Fortaleses: Et permet gestionar els teus propis esdeveniments, et proporciona estadístiques i igual que les altres l'accés es pot controlar a través de la lectura d'un codi QR.
 - Desavantatges: Esta pensada per esdeveniments mitjans o petits, ja que per un gran volum de dades pel control d'aforament i accés recomanen utilitzar una altre aplicació.

SWR-2.2: A cada treballador se l'assignarà un o més d'un rol.

SWR-2.3: El treballador ha de tenir un apartat on poder consultar les seves dades personals.

SWR-2.4: A l'apartat del treballador hi ha d'haver un lector de QR.

SWR-2.5: L'administrador ha de poder crear/eliminar treballadors, assignant-li totes les seves dades.

SWR-2.5: La informació dels usuaris a l'aplicació han de garantir la Llei Orgànica de protecció de dades.

SWR-2.6: Només el gestor de la Base de Dades podrà crear nous administradors de l'aplicació.

SWR-3: Tots els usuaris estaran lligats a un edifici.

SWR-3.1: Cada edifici estarà format per distintes seccions (pisos) on només podran entrar el personal autoritzat.

SWR-3.2: Cada pis disposarà de diverses sales accessibles només pel personal autoritzat.

SWR-3.3: Cada sala tindrà un material disponible.

SWR-3.4: Cada edifici, pis i sala tindran un codi QR lligat per garantir l'accés a aquell espai.

SWR-3.5: Els accessos als espais només es podran dur a terme durant l'horari d'oficina.

SWR-4: Els administradors poden generar històrics dels accessos i quedaran guardats a la Base de Dades.

SWR-5: L'aplicació ha de ser intuïtiva i no estar sobre-carregada.

SWR-6: L'aplicació es mostrarà en català, castellà o anglès segons l'idioma configurat al dispositiu mòbil, en qualsevol altre cas agafarà l'anglès com idioma principal.

També s'haurà de tenir en compte que només l'administrador podrà veure la informació de tots els usuaris, i un usuari només podrà accedir a la seva informació, ja que s'ha de respectar la Llei Orgànica de protecció de dades, i encara que no s'està treballant amb informació sensible s'ha de tenir en compte.

En fer insercions, eliminacions o edicions a la base de dades s'espera que la velocitat sigui fluida i no haver d'esperar-se massa temps en veure els canvis reflectits.

S'haurà de fer com a mínim *User Testing*, per a poder veure si l'aplicació és intuïtiva i trobar algun bug.

	Lectura QR	Històric	Diversos espais	Registre d'usuari
<i>Ticketea</i>	Si	No	Si	Si
<i>Atrápalo</i>	Si	No	Si	Si
<i>Enterticket</i>	Si	Si	No	Si

Taula 1 - Resum estat de l'art

Al analitzar el mercat actual de les aplicacions de control d'accés es podria dir que la lectura de QR per garantir l'entrada als espais és un dels requisits fonamentals, però podem trobar que no estan molt enfocades a la generació d'històrics, el qual és un punt important pel desenvolupament d'aquest projecte, és aquest el motiu pel qual s'ha decidit fer una aplicació des de zero.

2.2 Definició de requisits

A continuació es definiran els requisits que ha de satisfer l'aplicació de control d'accessos, aquests requisits poden anar variant durant el desenvolupament del projecte, ja que es tracta d'un desenvolupament àgil.

SWR-1: L'aplicació notificarà si l'usuari pot accedir a la zona a la qual esta intentant entrar.

SWR-2: A l'aplicació han de poder accedir dos tipus d'usuaris, administradors i treballadors.

SWR-2.1: Cada usuari ha de tenir un compte per poder accedir a l'aplicació.

2.3 Estimació de costos

Per a dur a terme el desenvolupament del projecte s'utilitzarà un ordinador valorat en 700€ que amortitzant-lo en 4 anys, i havent de fer un ús de 300 hores per a la realització del projecte aquest recurs en sortiria per 1€ més o menys. Si també es té en compte el preu del kW/hora s'han d'afegir 45€ més (tenint en compte que el preu del kWh es de 0.15€/kWh)

També seran necessaris els rols de Project Manager, desenvolupador, tester i un analista. A la següent taula es pot veure un resum dels costos dels rols:

Rol	Preu/hora	Hores treball	Total(€)
<i>Project manager</i>	24	10	240
<i>Developer</i>	18	200	3600
<i>Tester</i>	18	70	1260
<i>Analista</i>	20	20	400

Taula 2 - Resum preu/rol

Fent la suma de tots els recursos s'obté que el cost estimat del projecte és de 5.546€

3 DESENVOLUPAMENT

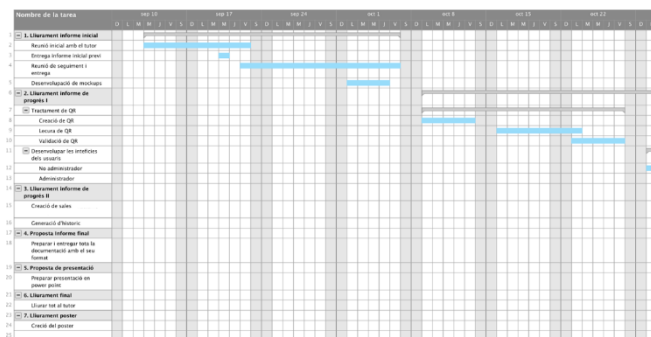
3.1 Metodologia i planificació de treball

La primera fase consisteix en diverses reunions, de forma presencial o via correu amb el tutor del projecte a desenvolupar per establir un objectiu inicial i a partir d'aquest punt començar amb el desenvolupament del projecte.

La fase de desenvolupament començarà amb la realització d'uns mock-up per començar a veure el disseny que hi haurà de prendre l'aplicació, aquests dissenys no seran definitius, però ajudaran també a veure si hi ha requisits que s'han de considerar que no s'havien vist abans. Un cop acabats es començarà a fer el desenvolupament de la base de dades i certs diagrames de casos d'ús per establir una bona base, un cop realitzats es començarà a desenvolupar l'aplicació a l'eina Android Studio. Aquesta fase de desenvolupament també inclourà una part de test per a garantir el bon funcionament de l'aplicació.

Finalment, un cop l'aplicació estigui acabada i sigui funcional, es realitzarà un vídeo de demostració del funcionament de l'aplicació.

A l'hora de realitzar la planificació s'ha creat un diagrama de Gantt amb les tasques a realitzar per poder anar assolint les fites correctament.



Il·lustració 1 - Diagrama de Gantt

Això implica una prioritització de requisits, és a dir, quins són els requisits més indispensables per poder començar a desenvolupar el projecte. El primer pas a fer és crear una base de dades per poder gestionar les dades que s'utilitzaran i emmagatzemaran a l'aplicació. Un cop la base de dades estigui ben definida s'implementarà a l'eina de desenvolupament. El següent pas serà realitzar la funcionalitat d'accés a l'aplicació i poder gestionar-los, és a dir, l'administrador podrà afegir o eliminar usuaris i en aquest mateix punt l'usuari només veurà informació sobre el seu perfil. A partir d'aquest punt es començarà amb el desenvolupament de la lectura del codi QR, l'usuari tindrà una opció on podrà escanejar el codi QR vinculat a la sala i obtindrà una resposta garantint-li o denegant-li l'accés. Finalment es desenvoluparà una opció pels administradors que els permetrà generar un històric dels accessos que hi ha hagut.

Com a requisit menys prioritari es podria dir que està el disseny de l'aplicació, ja que l'objectiu inicial és la funcionalitat, és a dir, l'aplicació mantindrà un nivell de simplicitat i en cas que donés temps es formataria una mica per fer un

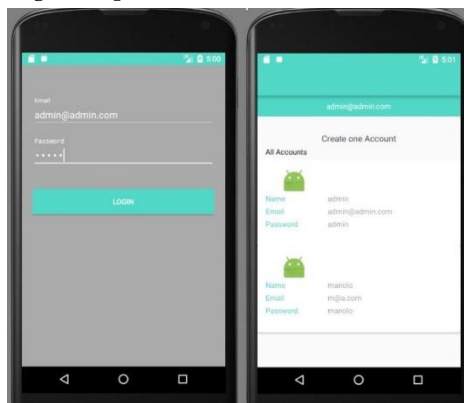
disseny més alegre.

3.2 Treball realitzat

Durant la segona fase de desenvolupament de l'aplicació els objectius que havia proposat i plantejat eren els de gestionar una interfície pels accessos dels usuaris de l'aplicació, generant un prototip i el seu posterior desenvolupament.

Per a poder realitzar el prototip el primer que es va fer va ser trobar una eina que deixés fer models d'aplicació mòbil per poder plasmar una primera idea de com serà l'aplicació i què hi haurà d'incloure. Durant la recerca es van considerar tres aplicacions, JustInMind, InVision i FluidUi. Després d'una prova bàsica amb les tres, JustInMind va ser l'estriada, ja que donava moltes més prestacions per a fer prototips d'aplicacions per a smartphone que les altres dues.

Un cop acabat el prototip es va passar al desenvolupament de la interfície pels accessos dels usuaris a l'aplicació, on ficant el seu correu i la seva contrasenya podien accedir a una llista amb tots els usuaris que contenia la base de dades, que podrem veure amb més detall en els següents punts.



Il·lustració 2 - Primer login en android

En acabar això el només tenir a la base de dades dels usuaris dificultava la continuació del desenvolupament, per tant es va fer un plantejament més extens de la base de dades. Això farà que canviï el codi desenvolupat, però part del que ha estat desenvolupat és podrà reaprofitar.

- **Base de Dades:** Aquesta base de dades ha de ser capaç d'emmagatzemar els usuaris de l'aplicació amb els seus rols assignats, a la següent taula podem veure els diferents rols que hi ha i quins accessos se'ls permetrà fer a les distintes sales.

Rol	Accessos permesos
Administrador	Cap
Cap empresa	Totes
Cap Projecte	Totes les vinculades amb treballadors d'oficina
Treballador d'oficina	La seva oficina
Treballador de la neteja	Totes

Taula 3 - Resum rol/Accés

També constarà de les sales disponibles a cada edifici i quin material s'hi pot trobar en elles. Finalment hi haurà un espai on es guardarà un històric dels accessos que s'han fet a l'edifici. A la secció A1 de l'apèndix es pot trobar un esquema d'aquesta base de dades [A1.1].

Aprofitant aquest assoliment de base s'hi han dissenyat també uns diagrames de casos d'ús amb el fi d'acabar de definir com funcionarà l'aplicació i poder trobar funcionalitats que no havien estat abans considerades. També es pot trobar un exemple a la secció A1 de l'apèndix [A1.2].

Els objectius plantejats per la tercera fase del desenvolupament de l'aplicació eren poder crear ja històrics amb la informació dels accessos i tenir ven desenvolupat l'accés per lectura de QR a la part de l'usuari. Al haver hagut una desviació en la planificació en la fita anterior aquests dos objectius no s'han pogut acabar d'assolir.

Durant la última fase de desenvolupament els objectius eren polir el codi de l'aplicació i fer testeig, però degut a la desviació de temps que hi havia i la incorporació d'altres tasques han fet que el temps per polir el codi i fer el testeig quedés reduït. Per una altre banda durant aquesta fase de desenvolupament s'han aconseguit els objectius per a fer l'aplicació funcional, és a dir, un usuari ja pot escanejar un codi QR des de el seu perfil, on se'l notificarà si pot accedir a aquell espai o no i tot això quedarà enregistrat en un històric al qual podrà accedir. A més a més els administradors ara poden gestionar les sales i poden accedir a l'històric global, on es veuen els accessos de tots el usuaris.

3.2.1 – Tecnologia de desenvolupament

Per al desenvolupament d'aquest projecte es va decidir treballar amb la tecnologia d'Android Studio.

Android Studio és un entorn de desenvolupament basat en IntelliJ IDEA, de la companyia JetBrains que ofereix funcions per al desenvolupament i la compilació d'aplicacions per dispositius Android. Aquesta eina et permet crear una estructura de treball en mòduls i fer-la més fàcil d'utilitzar, com es podrà veure en els següents punts a l'hora de la necessitat de comunicar *Activities* i *layouts*.

Uns dels principals motius per la decisió d'utilitzar aquesta tecnologia va ser perquè prèviament ja havia utilitzat l'eina i coneixia el seu funcionament, però mai havia fet un desenvolupament des de zero.

Per una altra banda la decisió també va ser tomada en base a quants clients potencials podria tenir l'aplicació, és a dir, si es desenvolupava per sistemes Android o iOS. Fent una cerca de la quantitat de dispositius venuts, els de sistema operatiu Android era notòriament més gran.

Sistema Operatiu	Quantitat en milers d'unitats	Percentatge
Android	327163,6	86.1%
iOS	51992,5	13.7%
Altres	821,2	0.2%
Total	379977,3	100 %

Taula 4 - Vendes mundials de dispositius mòbils

3.2.2 – Desenvolupament de l'aplicació

Per a poder utilitzar la base de dades dissenyada anteriorment a l'aplicació s'ha utilitzat l'extensió d'android "SQLiteOpenHelper", ja que no requereix un servidor per emmagatzemar la base de dades i és open source.

Per poder treballar amb les dades que volem gestionar el primer que s'ha de definir un esquema, que consisteix en la creació de models, és a dir, s'ha de crear una classe java per a cada entitat amb tots els atributs necessaris i crear les taules, com es podrà veure més endavant. També s'ha de crear una classe que serveixi com a contracte per a que guardi com a constant totes les característiques de la base de dades. Això es fa així, ja que la documentació d'android centrada en la gestió de base de dades ho recomana per evitar problemes a l'hora de treballar amb les dades.

A la definició de l'esquema a l'hora de crear taules s'han d'executar consultes de SQL amb la funció "execSQL" que proporciona la llibreria de SQLiteOpenHelper^[1].

Aquest fragment de codi és el que s'encarrega de crear l'entitat "Table.USER" que en aquest cas seria corresponent a l'entitat d'usuaris. Un cop es té la base de dades amb totes les seves entitats creades, el que s'ha de fer és passar la informació a un format que sigui insertable. També s'ha de tenir en compte que per llegir les dades s'ha de poder recórrer la base de dades i aprofitant el tipus de dades anterior podem transformar aquestes dades en objectes iterables.

Un cop la base de dades està llesta es passa al desenvolupament de l'accés a l'aplicació i de les interfícies d'usuari i d'administrador.

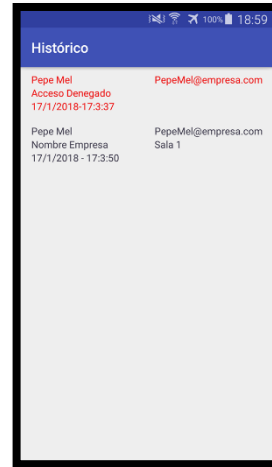
S'ha de començar creant una *Activity* que serveixi per fer l'accés a l'aplicació, és a dir, es crearà una classe java que accedirà a la base de dades comparant la informació d'aquesta amb la informació obtinguda dels camps emplenats. Android Studio ens permet generar també els *layouts* fent que sigui més senzill la gestió de les dades entre aquestes *layouts* i les *Activities*. Veient un exemple de codi es formen de la següent manera:

```

- Activity:
    inputEmail = (TextInputEditText) findViewById(
    ById(R.id.email);
- Layout:
    <android.support.design.widget.TextInputLayout
        android:id="@+id/til_email"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        an-
        droid:layout_marginTop="@dimen/activity_vertical
        _margin">

        <an-
        droid.support.design.widget.TextInputEditText
            android:id="@+id/email"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="@string/hint_email"
            android:inputType="textEmailAddress"
            an-
            droid:textAppearance="@style/TextAppearance.App
            Compat.Medium"
            tools:text="@string/hint_email" />

```



Il·lustració 3 - Històric d'usuari

La *Activity* d'administrador haurà de rebre la informació de l'*activity* anterior per a poder identificar que qui està accedint és un administrador. La vista inicial de la pantalla d'administrador és un menú de selecció per a determinar el que voldrà gestionar en aquell moment, es pot trobar el botó de gestió d'usuaris que en prémer-lo s'obrirà una nova *Activity* amb una llista amb tots els usuaris que conté la base de dades a més d'un botó flotant que serveix per afegir usuaris nous. En fer clic sobre un dels usuaris s'obrirà un *fragment* amb la seva informació i dos botons, un per eliminar i un altre per editar, en prémer el botó d'eliminar aquell usuari s'esborrarà de la base de dades i en prémer el d'editar s'obrirà un nou *fragment* amb tots els camps de l'usuari per poder editar-los. Aquest *fragment* és el mateix que es crea al polsar el boto flotant d'afegir usuari però amb la diferència que tots els seus camps estaran buits. Seguidament es troba el botó de gestionar les sales, on el funcionament és el mateix que l'explicat anteriorment però amb la diferència de que ara la informació que s'estarà mostrant és la relativa a les sales. Finalment existirà també un botó que permetrà veure tots els accessos que s'han enregistrat a la base de dades.

Per tal que l'aplicació no sembles que quedes aturada o penjada i per agilitzar els processos per tot el tema d'insercions, lectura i escriptura sobre la base de dades s'han creat unes classes asíncrones aprofitant la classe "AsyncTask^[3]" que ens proporciona java.

Per poder fer que l'aplicació es mostri en tres idiomes s'han de crear uns fitxers anomenats Strings que tindran un nom global i la seva definició en els 3 idiomes.

Al punt de resultats i a l'apèndix es poden veure més imatges representatives de l'aplicació.

3.2.3 – Bugs trobats

Ara com ara l'aplicació té els següents bugs que s'han de resoldre:

-Bg1: La imatge de perfil al layout d'usuari no s'adapta bé a l'espai donat.

-Bg2: Al *fragment* on l'administrador pot veure les dades d'usuari, no es carrega la imatge de perfil de l'usuari.

-Bg3: En fer la foto només la mostra per pantalla si el dispositiu està de forma horitzontal.

D'aquesta manera aconseguim lligar l'element del layout amb el que s'ha de desenvolupar a la *Activity*.

La *Activity* d'usuari haurà de rebre la informació de l'*activity* anterior per a poder identificar de quin usuari ha de mostrar la informació i està composta per una vista on es mostra la imatge de perfil l'usuari i les seves dades, a més d'un botó flotant que serveix per accedir a la càmera per poder fer l'escaneig d'aquest codi QR.

En polsar aquest botó s'accedeix a la càmera, què gràcies a una llibreria externa de Google anomenada "zxing" escanejarà la imatge i si troba el format de codi QR l'aplicació mostrarà si es pot o no accedir a l'espai desitjat.

Això internament funciona de la següent manera, en produir-se el reconeixement del codi QR la llibreria zxing generarà un toquen amb la informació que conté el QR i aquest toquen se'l passa a l'aplicació. En aquest cas el QR continuarà el nom de l'edifici o de la sala a la qual es vol accedir, ja que amb aquesta informació es podrà determinar si l'usuari té accés o no dins de la sala o edifici.

La determinació de si es té o no es té accés a l'espai ve donada, com s'ha pogut veure amb anterioritat, pel tipus de rol a les sales i per horari als edificis. Amb la creació de funcions com la funció *checkRoomAccessible*^[2].

Aquesta funció rep com a paràmetre el toquen generat anteriorment pel lector de codi QR i agafant la informació de l'usuari de la base de dades es pot determinar si l'usuari té accés.

A l'hora de notificar a l'usuari si ha pogut accedir es fa un bolcat d'informació a la base de dades consistent en l'hora i dia en què ha realitzat l'acció, qui l'ha fet i a on, aquesta és la informació amb la qual es generarà l'històric.

Finalment l'usuari podrà accedir al seu històric, on podrà veure una llista amb tots els accessos que ha anat fent, com es pot veure a la imatge que hi ha a continuació:

-Bg4: En fer clic al botó d'anar enrere quan estem a la càmera l'aplicació s'atura.

Resolució dels Bugs:

- Bg1: Es va canviar el format de rectangle a cercle i d'aquesta manera la foto ja sortia sencera.

- Bg2: Dins del layout "activity_user_detail" faltaven posar els paràmetres per poder emplenar la barra amb una imatge.

- Bg3: S'ha canviat la funcionalitat de l'escaneig i aquella funcionalitat ha desaparegut, per tant ja no es troba el bug.

- Bg4: S'ha canviat la funcionalitat de l'escaneig i aquella funcionalitat ha desaparegut, per tant ja no es troba el bug.

3.2.4 – Futures millores per l'aplicació

Una de les futures millores, pot ser la més important, podria ser mirar de posar la base de dades en un servidor, ja que amb la implementació actual les dades es guarden de manera local a la memòria del dispositiu on està instal·lada l'aplicació, potser l'ajuda d'un *web service* pot ser útil i en un futur faré una petita cerca per a veure com seria l'exportació.

Una altra millora, per a la part que utilitza l'administrador, seria la de crear un cercador, ja que es pot fer angoixós anar d'haver buscant els usuaris o les sales a la llista.

Si es mira la part d'usuari, potser una millora seria mantenir la sessió per no haver d'estar fent el login cada cop que es vol accedir o poder canviar la teva foto de perfil, ja que ara s'està posant per defecte una quan l'administrador crea l'usuari.

3.3 Problemes de metodologia i planificació

Durant el desenvolupament de l'aplicació es van començar a diversificar molt les coses, sense partir d'una base ben definida, és a dir, el desenvolupament es va centrar en l'objectiu que es tenia més proper segons la planificació inicial sense tenir en compte el global del projecte. Això lligat a un inexistent diagrama per establir la base de dades va suposar un canviar de les prioritats.

Un altre punt de la metodologia on hi ha hagut errors ha sigut a l'hora de no fixar un horari de treball, ja que en anar fent-ho en hores separades la productivitat baixa, per tant un nou punt inclòs a la nova metodologia de treball serà la fixació d'un horari per a continuar aquest projecte.

Aquets problemes trobats suposen una desviació respecte la planificació inicial, ja que al no considerar la creació d'una bona base de dades tot el procés d'implementació s'ha de moure cap a la següent fase de desenvolupament, això podria suposar que alguns dels objectius plantejats com a altres no es puguin dur a terme. Aquesta desviació també farà que el procés de desenvolupament es compacti més, ja que els objectius no assolits a la fita anterior és sumen als objectius de la fita actual.

Durant l'última fita de desenvolupament es realitzaran els objectius que no s'han obtingut encara per a asso-

lir el funcionament mínim de l'aplicació, a més l'administrador haurà de poder crear sales des de l'aplicació, tal com es fa amb els usuaris.

4 UTILITZACIÓ DE L'APLICACIÓ SEGONS USUARI

L'aplicació, com s'ha estat veient fins ara podríem dir que està dividida en 2 grans tipus d'usuaris, els administradors del sistema i els treballadors. Si bé com a treballador la cosa no és molt complicada, ja que la seva funció és accedir a l'aplicació amb el correu i la contrasenya que li proporcionaria l'empresa, escanejar el codi QR de la sala a la qual vol accedir i poder mirar el seu historial d'accessos el ser administrador té una mica més de complicació.

Com a administrador s'és responsable de portar tota la gestió de les sales i dels usuaris, a més de poder consultar l'històric d'accessos que hi ha hagut. A l'hora de crear o editar sales se'ls hi demanarà que omplin una sèrie de camps per poder gestionar aquesta sala, tals i com el nom de la sala, a quin edifici correspon, per qui és accessible^[4], la capacitat de la sala i el material que conté. Com el material pot ser molt variat haurà d'escriure una llista del material que es pot trobar dins d'aquella sala i haurà de seleccionar quina és la seva capacitat, ja que són coses que no estan predefinides i poden anar variant amb el pas del temps. Si es mira ara el tema dels usuaris es pot veure que la majoria de camps són informació sobre l'usuari, és a dir, que l'administrador només ha d'estriar el correu i la contrasenya, que com a recomanació es podria utilitzar el següent format per crear-les:

- Correu: NomCognom@empresaPertanyent.com
- Contrasenya: NomComgnom_X, on X pot se un número qualsevol.

En la següent imatge es pot veure un exemple:

Nombre	Joaquin
Apellidos	Campo Larriaga
Número de teléfono	667623465
DNI	03843850K
Correo	JoaquinCampo@empresa.com
Contraseña	JoaquinCampo_1
Rol	Software

Il·lustració 4 – Afegir/Editar usuari desde l'administrador.

5 TEST

Per a poder realitzar el test d'usuaris s'ha demanat ajut a unes persones dins d'un rang d'edat en el qual es veurien obligats a utilitzar aquesta aplicació, en l'estar enfocada per empreses el rang estriat ha estat entre els 65 anys i els 16.

Se'ls va passar un full amb les següents instruccions:

- Ets l'usuari amb correu "admin@admin.com" i contrasenya "Administrador_1". Crea, edita i elimina una sala. Crea, edita i elimina un usuari. Mira l'històric.
- Ets l'usuari amb correu "usuari@empresa.com" i contrasenya "Usuariproba_1". Intenta accedir a la sala amb el nom "Sala 1", digues si has pogut entrar i mira el teu historial d'accessos.

Els resultats van ser els següents:

Majoritàriament cap dels usuaris va tenir problemes a l'hora de realitzar les accions que se'ls van demanar, excepte un parell de punts que es veuran a continuació de la taula de resum del test.

	Problemes Part 1	Problemes Part 2	Intuitiva?
Persona de 50 anys	No	No**	Si
Persona de 23 anys	No*	No	Si
Persona de 18 anys	No	No	Si

Taula 5 - Taula resum Test d'usuari.

* A l'hora de la creació, en haver-hi d'assignar un rol a l'usuari aquesta persona va fer notar que si no l'hagués dit el tipus de rols que esperava l'aplicació ell no hagués estat capaç d'emplenar aquest espai correctament, per tant l'elaboració d'una guia per l'administrador és interessant^[4]. Per una altra banda va intentar eliminar l'usuari administrador, i a pesar d'haver estat eliminat l'aplicació seguia executant-se amb la seva informació, cosa que es podria evitar si es creessin sessions, com s'ha proposat a futures millores.

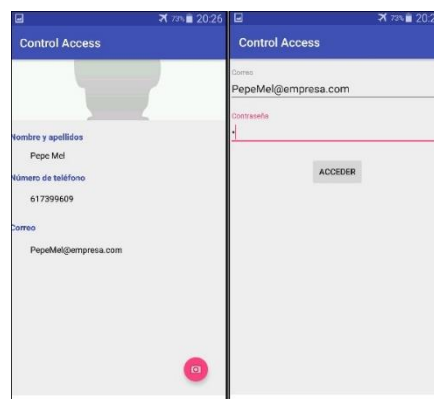
**A la part de veure l'historial d'accessos li va costar una mica associar que el text es podia clicar per accedir-hi.

6 RESULTATS

A la tercera fita l'aplicació funcionava sense problemes, però no estava tota la seva implementació feta.

Els usuaris podien accedir a l'aplicació, veure les seves dades i intentar escanejar un codi QR, però en fer-ho els deia que no s'havia trobat cap QR que coincideixi, ja que aquesta part no estava acabada d'implementar. Per una altra banda, els administradors, podien afegir, eliminar i editar usuaris, però encara no podien veure l'històric.

Imatges capturades de l'aplicació a la fita 3:



Il·lustració 5 - Vista d'usuari i login

El resultat final de l'aplicació és el següent:

- L'administrador pot veure una llista amb tots els usuaris i totes les sales disponibles. També pot accedir a l'històric per veure tots els accessos realitzats per tots els usuaris.
- L'administrador pot afegir, editar i eliminar sales i usuaris sense la necessitat de contactar amb els gestors de la base de dades.
- A més a més dels rols predeterminats els administradors podran crear altres tipus de rols que tindran la mateixa forma d'accés que els treballadors d'oficina.
- A cada sala poden afegir el material que ells desitgin i poden posar la capacitat de persones que poden accedir-hi, encara que no està controlat.
- L'usuari té accés al seu perfil on podrà veure les seves dades i la seva imatge de perfil.
- L'usuari pot fer un escaneig del QR associat a l'espai i es mostra un missatge on veu si pot accedir-hi o no.
- L'usuari té accés al seu històric d'accessos.

En resum, la funcionalitat bàsica de l'aplicació, que era la de determinar si un espai era accessible per un usuari amb la lectura d'un codi QR, ha estat aconseguit, a més s'ha implementat tota la part de gestió de les sales o espais a través de l'usuari administrador.

En el següent link podem trobar un vídeo del funcionament de l'aplicació i a l'apèndix s'hi mostren més captures:

<https://www.youtube.com/watch?v=91hdzV98Rvg>

7 CONCLUSIÓ

Durant la realització d'aquest treball he estat capaç de veure que una bona captura de requisits és molt important per saber quin és el producte final desitjat. Per a poder arribar-hi s'han d'establir molt bé unes bases de partida i prioritzar el treball a fer, ja que si no hi ha molta dispersió i no s'arribarà a l'objectiu.

També puc concloure que tenir un bon coneixement de la tecnologia amb la qual s'està desenvolupant l'aplicació pot ajudar a no desviar-se de la planificació, en el meu cas ja coneixia aquesta tecnologia, però mai havia estat jo sol desenvolupant en ella, això ha fet que tingui petites desviacions en la meua planificació, ja que en fer l'estimació

del temps que trigaria a fer una cosa era sempre molt optimista i el temps que acabava dedicant sempre era major al que jo havia planejat. Si ens ho mirem en forma de taula, per tenir una mostra més gràfica seria de la següent forma:

Tasca	T planejat(h)	T dedicat(h)	Desviació(h)
Base de Dades	15 h	24	9 h
Login	2 h	6 h	4 h
TOTAL	17 h	30 h	13 h

Taula 6 - Desviacions patides al desenvolupament

Per una altra banda cada cop que he estat treballant en l'aplicació he estat aprenent coses noves que poden ser-me útils en un futur.

Finalment l'objectiu a estat aconseguit a pesar de totes les desviacions en el temps que m'han anat sorgint.

En resum, tenir una idea clara del producte i poder definir la base de manera adequada ens farà ser capaços d'avançar de manera adient al nostre objectiu de la mateixa manera tenir una motivació per aprendre ajuda a seguir endavant encara que no es facin realitat tots els objectius.

AGRAÏMENTS

Els agraïments d'aquest treball són primer de tot per a Marc Talló, el professor que ha estat tutoritzant tot el treball durant aquests tres mesos, ja que és gràcies a la seva ajuda i punts de vista que fa que això segueixi cap endavant i per la família, ja que són un punt de recolzament que sempre està allà. També vull fer menció dels meus amics de tota la vida, ja que han sigut una font d'antiestrès.

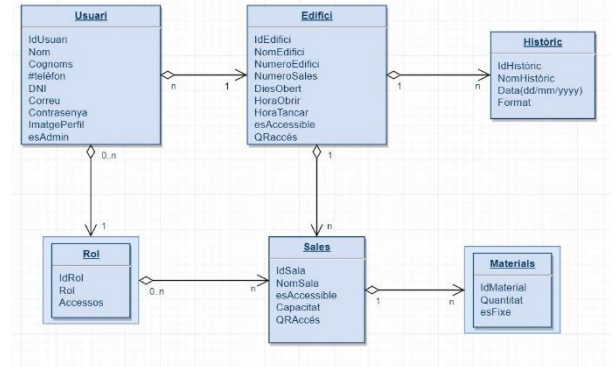
BIBLIOGRAFIA

- [1] Ticketea Checkpoint [Consulta Septembre 2017]. Disponible a: <https://www.ticketea.com/producto/checkpoint/>
- [2] Control de Accesos Atrápalo [Consulta Septembre 2017]. Disponible a: <https://www.atrapalo.com/>
- [3] Enterticket [Consulta Septembre 2017]. Disponible a: <https://www.enterticket.es/?page=prepara>
- [4] Creació diagrames de Gantt [Consulta Septembre 2017] Disponible a: <https://app.smartsheet.com/>
- [5] Programació android [Consulta Gener 2018] Disponible a: <https://developer.android.com/guide/index.html>
- [6] Just In Mind [Consulta a Octubre 2017] Disponible a: <https://www.justinmind.com/>
- [7] Stack Overflow [Consulta a Gener 2018] Disponible a: <https://stackoverflow.com/>
- [8] Gartner [Consultat Septembre 2017] Disponible a: <https://www.gartner.com/technology/home.jsp>
- [9] QR code Generator [Consulta a Gener 2018] Disponible a: <http://es.qr-code-generator.com/>

APÈNDIX

A1. IMATGES SUPLEMENTARIES

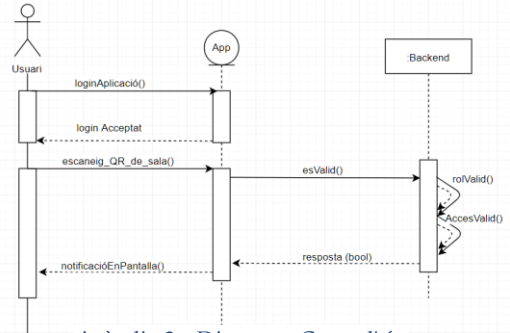
A1.1 - Base de dades de l'aplicació:



Apèndix 1 - Diagrama ER

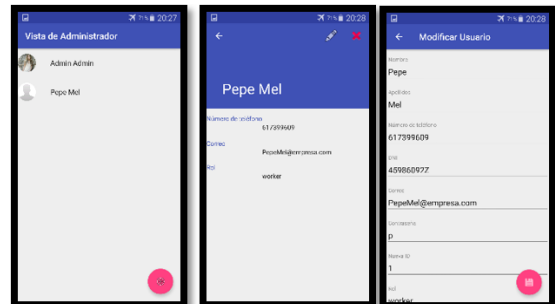
A1.2 - Diagrama de casos. Accés a sala:

L'usuari entrarà a l'aplicació i farà ús de la funcionalitat d'escaneig de QR per veure si pot accedir a la sala.

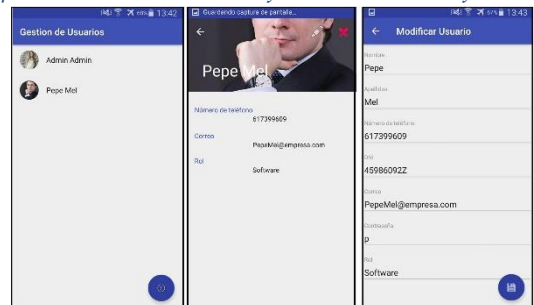


Apèndix 2 - Diagrama Casos d'ús

A1.3- Més imatges de l'aplicació:

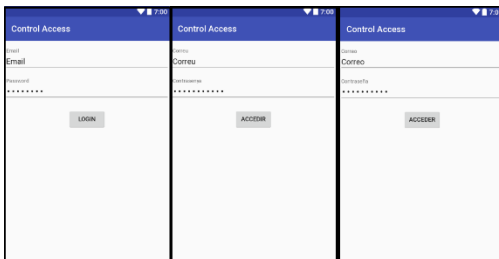


Apèndix 3 - Vistes de la modificació d'usuari a la fita 3



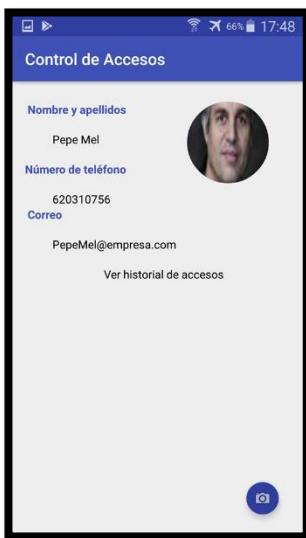
Apèndix 4 -Vistes de la modificació d'usuari versió final

A1.4 – Diferència d'idiomes en l'aplicació:



Apèndix 5 - Login en 3 idiomes

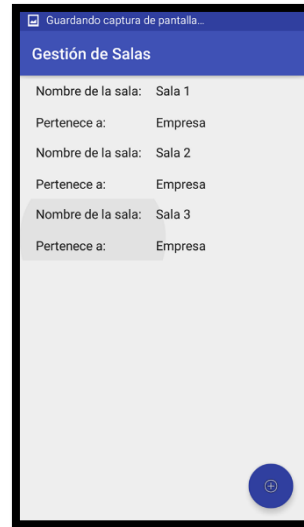
A 1.5 – Imatges capturades de l'aplicació:



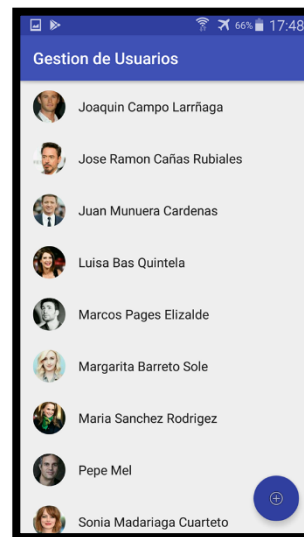
Apèndix 4 - Perfil d'usuari



Apèndix 5 - Pantalla inicial administrador



Apèndix 8 - Llista de sales



Apèndix 9 - Llista d'usuaris

A2. GLOSSARI

QR: És una matriu de dues dimensions formada per una sèrie de quadres negres sobre un fons blanc.

Mock-up/prototip: És un disseny esquemàtic de com serà l'aplicació que s'està desenvolupant. Aquest disseny pot incloure també part de la funcionalitat.

Classe java: És un fitxer que conté codi en java.

Open Source: Software distribuït i desenvolupat de manera lliure.

Activity: És un component de l'aplicació que conté una pantalla amb la qual els usuaris poden interactuar per realitzar una acció.

Layout: Són els formats de les vistes que contindrà l'aplicació.

Fragment: Són una part de la interfície d'usuari que es poden carregar sobre un *Activity*. És una secció modular d'una *Activity*.

Web Service: És una tecnologia que utilitza un conjunt de protocols i estàndards que serveixen per intercanviar

dades entre aplicacions.

User Testing: És una tècnica per avaluar un producte mitjançant proves que fan els mateixos usuaris.

A3. REFERÈNCIES

[1] Exemple de l'ús de la funció `execSQL`:

```
db.execSQL("CREATE TABLE " + Table.USER + "("
    + BaseColumns._ID + " INTEGER PRIMARY
KEY AUTOINCREMENT,"
    + User.ID + " TEXT NOT NULL,"
    + User.NAME + " TEXT NOT NULL,"
    + User.SURENAME + " TEXT NOT NULL,"
    + User.PHONE + " TEXT NOT NULL,"
    + User.DNI + " TEXT NOT NULL,"
    + User.EMAIL + " TEXT NOT NULL,"
    + User.PASSWORD + " TEXT NOT NULL,"
    + User.IS_ADMIN + " TEXT NOT NULL,"
    + User.PROFILE_PICTURE + " TEXT,"
    + "UNIQUE (" + BaseColumns._ID + ")");
```

[2] Funció per comprovar si una sala és o no accessible:

```
public boolean checkRoomAccessible(IntentResult result){if(database.checkRoomName(result.getContents().toString())if(database.getRoomsData(result.getContents().toString()).getNameRoom().equals(result.getContents())&& (database.getRoomsData(result.getContents().toString()).getIsAccessible().contains(database.getUserData(mailFromLogin).getIsAdmin()))| | database.getUserData(mailFromLogin).getIsAdmin().equals("Boss")| | database.getUserData(mailFromLogin).getIsAdmin().equals("Cleanup")| | (database.getUserData(mailFromLogin).getIsAdmin().equals("Project Manager")&&(database.getRoomsData(result.getContents().toString()).getIsAccessible().contains("Software")| | (database.getRoomsData(result.getContents().toString()).getIsAccessible().contains("Hardware"))))){return true;}else{return false;}}else{return false;}}
```

[3] Exemple amb codi de la funció d'AsyncTask.

```
private class GetRoomByIdTask extends AsyncTask<Void, Void, Cursor>{

    @Override
    protected Cursor doInBackground(Void... params) {
        return database.getRoomById(mRoomID);
    }

    @Override
    protected void onPostExecute(Cursor cursor) {
        if (cursor != null && cursor.moveToLast()) {
            showRooms(new Rooms(cursor));
        } else {
            showLoadError();
        }
    }
}
```

```
        getActivity().setResult(Activity.RESULT_CANCELED);
        getActivity().finish();
    }
}
```

Si ens fixem en `GetRoomByIdTask` podem veure que té una funció que s'executa de fons on fa una consulta a la base de dades per poder extreure la informació de totes les sales existents i en fer la crida de l'execució d'aquesta classe ja tindrà precarregades les dades necessàries i podrà operar de manera eficaç sense fer esperar a l'usuari.

```
private class AddEditRoomsTask extends
AsyncTask<Rooms, Void, Boolean> {

    @Override
    protected Boolean doInBackground(Rooms... params) {
        if (mRoomID != null) {
            return database.updateRooms(params[0],
mRoomID) > 0;
        } else {
            return database.saveRoom(params[0]) > 0;
        }
    }

    protected void onPostExecute(Boolean result) {
        showRoomsScreen(result);
    }
}
```

Si ens fixem ara en `AddEditRoomsTask` la idea és la mateixa, però canvia que en comptes de precarregar tota la llista de sales de la base de dades només precarregarà la informació d'una sala, a la que s'haurà pressionat per editar en aquest cas.

[4] Explicació dels rols.

Rol	Català	Castellà	Angès
Administrador	admin	admin	admin
Cap empresa	Cap	Jefe	Boss
Cap Projecte*	ManagerDe-Projecte	ManagerDe-Proyecto	ProjectManager
Treballador d'oficina**	-	-	-
Treballador de la neteja	Netejador	Limpiador	Cleanup

* Si a la sala hi poden accedir treballadors d'oficina amb els rols de Hardware o Software ells també podran accedir-hi.

** Poden crear tants tipus de treballadors com desitgin, sempre que els vinculin a alguna sala, o mai podrien accedir a cap.