

## Treball final de grau

**Estudi:** Grau en Enginyeria Informàtica

**Títol:** Desenvolupament de millores al sector logístic:  
Tecnologia NFC

**Document:** Memòria

**Alumne:** Marc Costa Quer

**Tutor:** Antonio Bueno Delgado

**Departament:** Arquitectura i tecnologia de computadors

**Àrea:** Arquitectura i tecnologia de computadors

**Convocatòria (mes/any):** Setembre 2020

---

# Agraïments

Primer de tot m'agradaria mostrar agraïment a totes la meva família, per haver-me ajudat durant tots aquests anys de carrera a la universitat.

Als meus companys de la universitat, per tots els moments viscuts, els treballs conjunts, i amb l'ajuda que ens hem fet tots per anar tirant endavant projectes difícils.

Al meu tutor Antonio Bueno, per ajudar-me i creure en el meu projecte des d'un principi, i guiar-me per no sortir-me dels objectius, també l'atenció feta durant les vacances d'Agost.

I finalment a tots els professors que s'han preocupat per ensenyar-nos els diferents conceptes vistos a les seves assignatures, resolent molts de dubtes a les tutories quan es necessitava.

A tots ells gràcies.

# Índex

1	Introducció . . . . .	<b>6</b>
1.1	Motivació . . . . .	6
1.2	Propòsit i objectius . . . . .	6
2	Estudi de viabilitat . . . . .	<b>7</b>
2.1	Introducció E-commerce . . . . .	7
2.2	Mercat dels dispositius mòbils . . . . .	9
2.3	Codis de barres vs Chips NFC . . . . .	10
3	Metodologia . . . . .	<b>11</b>
4	Planificació . . . . .	<b>12</b>
4.1	Planificació inicial amb diagrama de Gantt . . . . .	12
4.2	Planificació final amb diagrama de Gantt . . . . .	12
5	Marc del treball i conceptes previs . . . . .	<b>13</b>
6	Requisits del sistema . . . . .	<b>14</b>
6.1	Requisits de l'aplicació WareHouse-Sync . . . . .	14
7	Estudis i decisions . . . . .	<b>15</b>
7.1	Primera Idea: Supermartek . . . . .	15
7.1.1	Explicació i evolució de l'idea . . . . .	15
8	Anàlisi i disseny del sistema . . . . .	<b>19</b>
8.1	Cas d'ús de WareHouse-sync . . . . .	19
8.2	Casos d'ús individuals . . . . .	20
8.3	Esbós de les pantalles de l'aplicació . . . . .	23
9	Implementació i proves . . . . .	<b>25</b>
9.1	Primera Idea: Supermartek . . . . .	25
9.1.1	Disseny i implamantació del prototip . . . . .	25
9.1.1.1	Material necessari . . . . .	25
9.1.1.2	Connexió de components . . . . .	26
9.1.1.3	Programació de l'arduino . . . . .	27
9.1.1.4	Configurar PLX . . . . .	28
9.1.1.5	Entrar codi producte . . . . .	28
9.1.1.6	Entrar chips rfid dels productes . . . . .	29
9.1.2	Objectius . . . . .	29
9.2	Segona Idea: WareHouse-Sync . . . . .	30
9.2.1	Explicació estructura del projecte Android Studio . . . . .	30
9.2.2	La base de Dades en local (ROOM) . . . . .	31
9.2.3	Disseny de les activitats de l'aplicació . . . . .	33
9.2.3.1	LoginActivity.java . . . . .	33
9.2.3.2	scrollingActivity.class . . . . .	35
9.2.3.3	RegistrarProducteActivity.java . . . . .	37
9.2.3.4	NfcReaderActivity.java . . . . .	38
9.2.3.5	ProductesActivity.java . . . . .	41
9.2.3.6	PrestatgesActivity.java . . . . .	42
10	Implantació i resultats . . . . .	<b>43</b>
10.1	Resultats: Demostració dels requeriments . . . . .	43

---

10.1.1	Instal·lació . . . . .	43
10.1.1.1	Log-in . . . . .	45
10.1.1.2	Visualitzar magatzem, fer scroll i visualitzar el menú . . . . .	46
10.1.1.3	Prèmer una estanteria per veure els prestatges . . . . .	47
10.1.1.4	Llegir xip NFC . . . . .	48
10.1.1.5	Registrar un paquet i buscar paquet . . . . .	50
10.1.1.6	Localitzar paquet en el magatzem . . . . .	52
10.1.1.7	Esborrar un paquet del magatzem . . . . .	53
10.1.1.8	Sortir de l'aplicació . . . . .	54
11	Conclusions . . . . .	<b>55</b>
12	Treball futur . . . . .	<b>56</b>
13	Annex . . . . .	<b>57</b>
13.1	Manual d'usuari . . . . .	57
13.1.1	Pantalla de login . . . . .	57
13.1.2	Magatzem . . . . .	58
13.1.3	Menú de l'aplicació . . . . .	59
13.1.4	Registrar paquet . . . . .	60
13.1.5	Buscar paquet . . . . .	61
13.1.6	Eliminar paquet . . . . .	62
13.1.7	Localització del paquet . . . . .	63
	Referències . . . . .	<b>64</b>

## Índex de figures

1	Gràfic del creixement dels usuaris del E-commerce [1]	7
2	Gràfic de les botigues online més utilitzades a Espanya [2]	8
3	Gràfic de les vendes a tot el món de dispositius mòbils [3]	9
4	Diagrama Gantt d'objectius/setmana inicial.	12
5	Diagrama Gantt d'objectius/setmana final.	13
6	Logo del primer projecte Supermartech.	15
7	Diagrama cas d'ús general	19
8	Esbós de les pantalles de l'aplicació	23
9	Esbós de les pantalles de l'aplicació 2	24
10	Pins arduino UNO utilitzats.	26
11	Esquema elèctric del chip RFID.	26
12	Prototip inicial supermartech	27
13	Pantalla PLX configuració.	28
14	Botonera del prototip.	29
15	Estructura projecte Android Studio.	30
16	Esquema de crides de Room i l'aplicació.	31
17	Esquema de la BBDD de l'aplicació.	31
18	Mode de desenvolupador activat.	44
19	Selecció del dispositiu a l'Android Studio	44
20	Pantalla de log-in de l'aplicació.	57
21	Pantalla principal on es veu el mapa del magatzem.	58
22	Pantalla de menú de WareHouse-Sync	59
23	Vista de buscar i seleccionar un paquet.	61
24	Pantalla per eliminar un paquet del magatzem.	62

# 1 Introducció

## 1.1 Motivació

Des de fa molts anys que en la ment tenia la idea d'automatitzar i millorar els codis de barres d'un supermercat. Gràcies a l'augment exponencial del mercat dels dispositius mòbils dels darrers anys, han abaixat molt els preus de les tecnologies que porten incorporada en ells. En el meu projecte tractaré la tecnologia NFC com a pilar central i a poder ser incorporada en els dispositius mòbils, per fer més accessible i senzill la utilització.

Ha estat un encert esperar uns anys per desenvolupar aquesta idea de modernitzar el sistema del codi de barres, el sector de l'e-commerce també està pujant any rere any, augmentant les empreses de transport, per tant fent créixer el món de la logística que és on el meu projecte entra.

En cap projecte de la carrera l'he pogut desenvolupar, ja que tots els treballs fets fins ara eren molt pautats. En el projecte de final de carrera era el millor moment per portar a la llum i intentar fer d'una idea un bon projecte i a poder ser un producte amb cara i ulls a l'hora de vendre en el sector de la logística per facilitar el seu treball de gestió dins del sector amb la tecnologia del NFC.

Aquests avantatges tecnològics gràcies al mercat evolutiu dels smartphones i l'alt creixement en empreses de transport fan el meu projecte molt interessant i amb un alt potencial econòmic, ja que són mercats en alça i una millora en el sector podria ser molt fàcil de vendre i obtenir clients finals del producte.

Descrits els mercats on està enfocat el meu projecte, procediré a investigar petites i mitjanes empreses de logística comarcals, per tal de veure el seu funcionament, possibles millores en certs processos, estudiar la viabilitat d'algun producte que aportés millora a les empreses, a posteriori dissenyar i desenvolupar la idea, implementar-la, i revisar que el sistema sigui efectiu.

## 1.2 Propòsit i objectius

Les principals parts per assolir tot el desenvolupament del projecte són tres, com s'ha descrit abans s'haurà d'estudiar i investigar el sector de la logística a Girona, anant a diferents empreses per veure el seu funcionament, la distribució del magatzem, i aconseguir diferents opinions d'operaris del sector sobre el producte a desenvolupar, si el veuen viable i útil, pros i contres, etc.

1. Estudiar el perquè la tecnologia sobre el que es desenvolupa el meu projecte, NFC/RFID, aporta una millora sobre el codi de barres utilitzat des de fa anys.
2. Dissenyar una APP pel mòbil/tauleta que ajudi a agilitzar el procés d'emmagatzematge de les empreses amb la tecnologia NFC i que aportí millores dins del sector actual, estudiar la viabilitat del sistema desenvolupat per tal de ser un producte fàcil de vendre i tot el que comporta el canvi de tecnologia.
3. Finalment, implementar un prototip del sistema final, per demostrar el correcte funcionament i els avantatges d'utilitzar-lo envers els que s'utilitzen actualment com el codi de barres.

M'agradaria remarcar que la idea inicial del projecte era automatitzar el fet de "passar per caixa" en un supermercat, i s'ha desenvolupat un prototip per millorar l'eficiència d'aquest procés, però pel simple fet de no haver assolit els objectius que volia aconseguir d'aquest prototip, m'he fet repensar la idea del treball, dissenyant una aplicació per Android per gestionar les entrades de paquets a un magatzem, i poder localitzar-los fàcilment, tot això s'explica amb detall en apartats posteriors.

---

## 2 Estudi de viabilitat

### 2.1 Introducció E-commerce

El creixement massiu d'internet des de fa dècades ha potenciat molt els negocis virtuals, comunicacions a llarga distància, informació a l'instant facilitant l'aprenentatge i ens ha donat diferents formes d'organitzar-nos sense moure'ns de casa.

Gràcies als diferents avantatges, i les noves empreses que veuen el potencial d'internet sorgeixen empreses molt importants dins del sector tecnològic, anys enrere va sorgir una de les empreses més importants actualment, Amazon, va crear una revolució en la compra/venda per internet i massificant el E-commerce a tot el món.

Com és pot veure en el gràfic, el sector de la compra/venda en línia any a any hi ha més usuaris, i té una previsió de creixement en els anys vinents. Fent aquest mercat molt suculent per noves empreses i projectes de cara a intervenir i aprofitar la pujada del sector.(Figura 1)

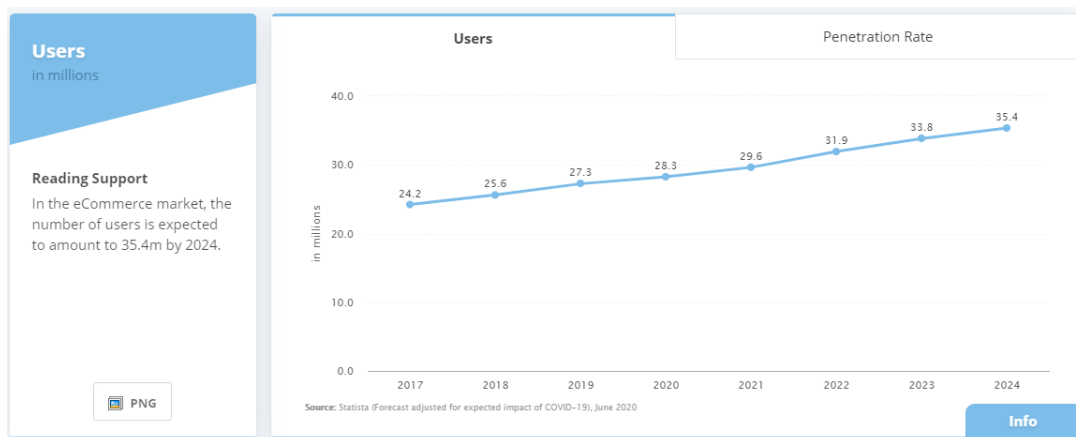


Figura 1: Gràfic del creixement dels usuaris del E-commerce [1]

També podem veure que cada any es compra més productes de tots els tipus via internet, i fent baixar el comerç físic que coneixíem fins ara, obrint noves oportunitats de negoci.

Si comprovem les empreses més importants del sector E-commerce a Espanya podem veure que les més importants, i ja especialitzades en això són Amazon i AliExpress. Mirant amb més atenció podem veure que moltes són botigues físiques que ara s'han reinventat obrint també la seva pàgina web per vendre els productes en línia i aprofitant el mercat en alça. Veiem tota mena de botigues com supermercats, botigues de tecnologia, cosmètica i fins i tot mobles.(Figura 2)

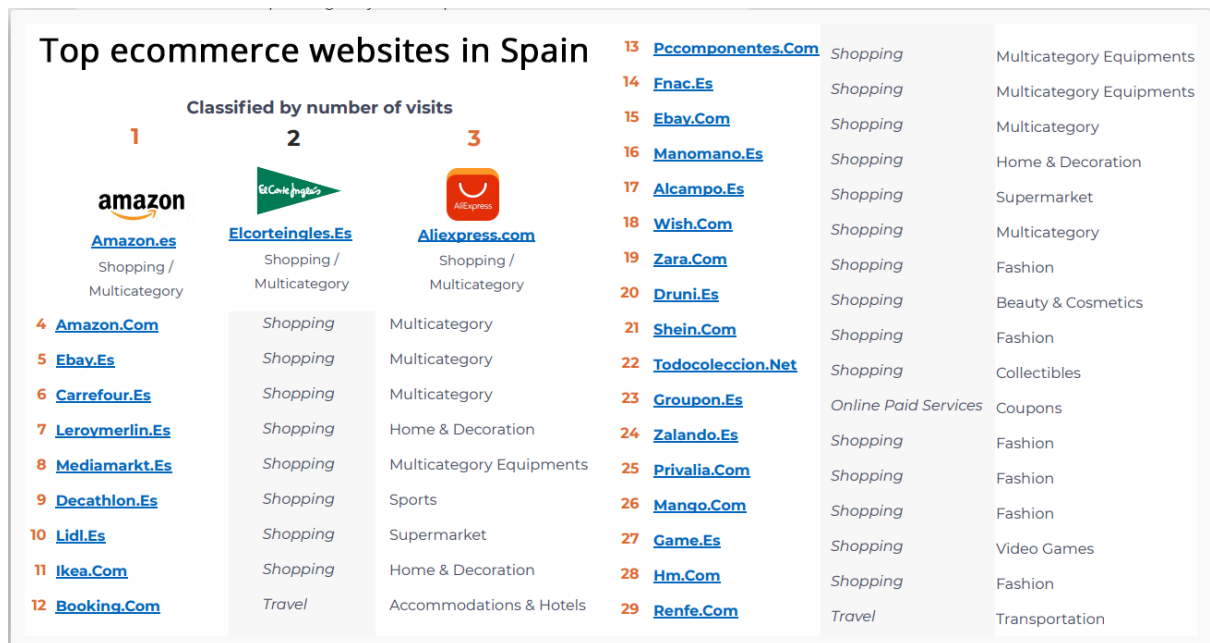


Figura 2: Gràfic de les botigues online més utilitzades a Espanya [2]

El fet que moltes botigues de caràcter físic també es reinventin cap a una nova manera de vendre els seus productes, és un fet que el mercat del E-commerce s'està tornant dels més importants avui en dia, ja que proposa una facilitat pels clients i molta comoditat, ja que des de casa pots comprar i directament una empresa de transport t'ho portarà al teu domicili sense haver de preocupar-te.

Aquí és el punt fort del projecte, ja que el sector del E-commerce està pujant cada any i té previsió d'augmentar en els següents anys, estan sortint moltes empreses noves de Logística per fer arribar la gran demanda de paquets comprats via web, als domicilis corresponents. Si aconseguim agilitzar el procés d'emmagatzematge i ajudar amb l'organització del magatzem pot sortir un sistema molt fàcil de vendre a les diferents empreses noves del sector, ja que els hi facilitarà la gestió i podran servir els productes més ràpid.



---

## 2.2 Mercat dels dispositius mòbils

En aquests darrers anys el mercat dels dispositius mòbils també porta tendència en créixer, gràcies a totes les aplicacions que et permet fer un Smartphone avui en dia no trobaràs cap persona que no en tingui un. Moltes persones fins i tot han creat dependència al seu dispositiu, et permet està connectat tot el dia a internet, fer fotos en el moment just, executar tot tipus d'aplicacions, o simplement et permet mirar l'hora. Observant el següent gràfic veient l'augment en Milions de vendes de cada any en el món.(Figura 3)

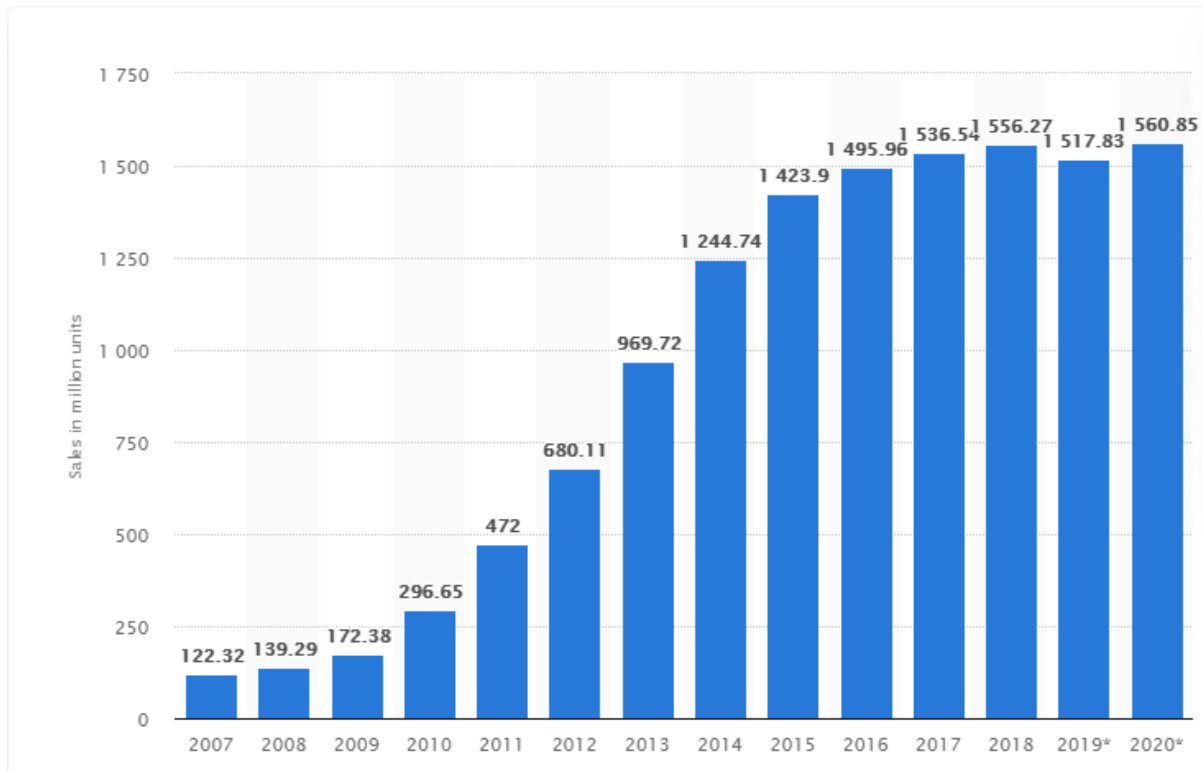


Figura 3: Gràfic de les vendes a tot el món de dispositius mòbils [3]

La pujada d'aquest mercat ens ofereix una gran possibilitat d'aprofitar el hardware i les aplicacions d'un smartphone per tal de fer un producte per les empreses de logística, que millori l'eficiència dels seus processos per tal de facilitar la feina als operaris, i centrar-nos en 2 mercats en alça com són el de la logística i el dels dispositius mòbils.

---

## 2.3 Codis de barres vs Chips NFC

La principal idea del projecte és estudiar la manera de fer un canvi de tecnologia millor a la dels codis de barres utilitzats des del 1952, una tecnologia eficient, antiga, però que necessita una renovació per fer diferents processos més eficients del que es poden fer ara.

L'aposta de millora del codi de barres és la tecnologia NFC, poc utilitzada per les millores que aporta a molts aspectes, cada cop hi ha més processos que l'utilitzen com per exemple pagar des del mòbil sense haver d'utilitzar la targeta de crèdit.

En aquest punt ens centrarem a comparar les 2 tecnologies per demostrar que el canvi és a millor:

1. El principal avantatge que tenen els chips NFC respecte el codi de barres, és la traçabilitat que té, és a dir que un xip NFC/RFID té un codi individual per cada producte, en canvi el codi de barres utilitza el mateix codi per tot aquell tipus de producte, o lot de productes, etc.  
Molt útil en cas de tenir productes diferents com per exemple un centre logístic, que gestiona paquets i cap d'ells és igual, o també per tenir constància de pèrdues de productes, robatoris...
2. Els xips NFC poden ser escrits a part de ser llegits, i poden guardar molta més informació que un simple codi de barres o d'un Codi QR la qual cosa fa que no poden ser modificats un cop creats, en la tecnologia NFC plantejada podríem guardar-hi la data que caduca el producte i d'aquesta manera seria molt més fàcil saber si el producte està caducat, el preu, la data d'entrega, a on ha estat fabricat i altre informació important.
3. El codi de barres està molt limitat a l'hora de llegir la seva informació, ja que necessitem línia de visió amb el codi de barres, i que estigui col·locat de manera horitzontal perquè el detector làser el llegeixi bé, pot afectar que l'etiqueta estigui una mica xafada o bruta. Els xips NFC no cal la línia de visió esmentada, ni la col·locació per ser llegit, podem tenir diversos productes dins d'una caixa i llegir els xips NFC sense tenir línia de visió.
4. El temps de vida que pot tenir un xip RFID és molt llarga respecte a un codi de barres, pot aguantar més condicions ambientals i d'entorn. El codi de barres són fràgils respecte a molts entorns que ens podem trobar a fàbriques o magatzems, si aquests es troben en llocs humits.
5. La facilitat de lectura dels xips NFC permet la llibertat de què no fa falta una persona per llegir-lo, d'aquesta manera en un magatzem amb les estanteries molt altes, el carretó que va a buscar els paquets pot portar incorporat el lector RFID, i no cal l'ajuda d'una persona com es necessita en els codis de barres. Avantatge que ens abarateix el tractament dels xips NFC i siguin més barats, respecte a una etiqueta impresa, ja que no es necessita un operari que els vagi llegint.
6. A l'hora de fer inventari en un magatzem, la tecnologia NFC és molt més ràpida que utilitzar els codis de barres.
7. La falsificació dels xips NFC té un % molt baix, ja que com s'ha dit abans cada xip només té un codi propi, en canvi amb qualsevol impressora làser és infinitament més fàcil falsificar un codi de barres.
8. La tecnologia del codi de barres quan llegeixes una tira llarga de productes, hi ha moltes probabilitats de què es deixi per llegir un producte, per això al supermercat sempre hi ha el soroll quan passem cada producte, els xips NFC tenen un % d'incert quasi del 100% encara que fem tirades molt llargues de productes.

Totes aquests avantatges són les més importants de cara a vendre aquesta evolució de la tecnologia per referència de productes o paquets. Es veu una millora considerable, amb aquest projecte volem aprofitar aquests avantatges per fer un prototip utilitzant a favor alguna d'elles i començar introduir més els xips NFC al sector de la logística.

---

### 3 Metodologia

Quan vaig plantejar l'enfocament del projecte, vaig proposar-me 3 grans punts per assolir i portar una organització. Com s'ha dit anteriorment el primer punt és plantejar un dispositiu de tal manera que pogués vendre dins del sector de la logística i aprofitar totes els avantatges que aporta els xips NFC respecte als codis de barres. Dedicar unes hores desenvolupant la idea del supermercat, i després anar a diferents supermercats per preguntar què els hi semblava la idea o si la veien viable o no.

La idea era dispositiu per detectar xips NFC a una distància de 20cm, que ens milloraria el fet de "passar per caixa" en un Supermercat, cada producte portaria el seu xip NFC substituint el codi de barres, de tal manera que passant els productes pel detector ja ens sortiria la conte, i seria molt més ràpid que anar buscant i passant el codi de barres de cada producte.

El dispositiu no va assolir els objectius proposats prèviament perquè fos un producte a vendre i capaç d'agilitzar el procés de "passar per caixa", vaig estar uns dies replantejant la idea del projecte i pensant una altre idea.

Per culpa de la pandèmia vaig haver de girar tot el previst, així que vaig intentar contactar amb diferents empreses de Girona per anar a veure com tenien gestionat el magatzem, proposant el primer intent de millora del codi de barres, però m'hagués agradat portar-si un producte final, com és la segona idea que he desenvolupat. Pel poc temps que tenia a l'hora de quedar amb les empreses amb la pandèmia al cim nostre, vaig haver d'anar-hi a correu sense cap producte important

Un cop visitades 4 empreses de Girona observant com tenien organitzats els magatzems, fent diverses preguntes en els encarregats, vaig fer pluja d'idees i seleccionar la idea que em va semblar més bona per desenvolupar un producte final.

Així que el projecte va patir un replantejament del producte a desenvolupar i agafant un altre rumb, però mantenint l'objectiu principal que era millorar el codi de barres, en aquest cop més enfocat empreses de logística i no tant a supermercats.

Finalment l'idea que he desenvolupat per vendre a les diferents empreses, és una app on pots visualitzar el magatzem, i pots registrar el paquet amb els xips NFC de manera molt fàcil a una estanteria del magatzem. Mantenint la BBDD organitzada, permeten veure la posició dels paquets de forma visual sobre el mapa del magatzem, i essent molt fàcil i ràpida d'utilitzar. Donant una bona utilitat de cara a magatzems grossos o mal organitzats. Simplement amb el teu mòbil pots portar el control de la base de dades, i registrar els paquets amb el seu codi NFC.

Per desenvolupar l'aplicació primer vaig posar-me a fer un estudi dels requeriments, dissenyant les diferents pantalles per tenir totes les idees organitzades facilitant el procés de programar-la. S'utilitzaria l'Android Studio com a IDE per fer l'aplicació Android.

Un cop desenvolupat tota l'aplicació amb els requeriments previstos, simplement vaig haver de fer la documentació, deixant en constància tot el procés fet en el projecte, i explicant les parts més rellevants.

---

## 4 Planificació

La planificació és semblant a l'explicada en el punt anterior, on en el desenvolupament i pluja d'idees és el que esperava passar més temps del projecte, i efectivament han estat els punts més durs. Obtenir una bona idea i dur-la a terme amb tot el que comporta, pràcticament tot el temps del projecte ha recaigut sobre aquests apartats.

M'hagués agradat anar un cop desenvolupada l'aplicació poder anar a les empreses a fer una petita enquesta de què els hi semblava, però l'any no ha acompanyat.

Per una bona documentació també era un dels punts que volia dedicar-hi més temps, que és realment on es pot plasmar tot el projecte real fet.

### 4.1 Planificació inicial amb diagrama de Gantt

Per planificar-me la feina he utilitzat un diagrama de Gantt, m'agrada molt planificar amb aquests diagrames, portar el projecte i la feina organitzats facilita molt el treball a fer ajudant a assolir els objectius setmana a setmana. El treball el vaig començar al 9 de Març, però l'Abril no vaig poder seguir la planificació degut a les circumstàncies que ens trobavem.

La planificació inicial amb el diagrama de Gantt és el següent:

Activitat/ Setmana	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Pensar idea	✓	✓	✓	✓													
Contactar/Visitar Empreses			✓	✓	✓	✓								✓	✓		
Dememar Material Prototip			✓	✓	✓	✓											
Muntatge del prototip					✓	✓	✓	✓	✓	✓	✓	✓	✓				
Documentació								✓	✓	✓	✓	✓	✓	✓	✓	✓	
Preperar Presentació															✓	✓	✓

Figura 4: Diagrama Gantt d'objectius/setmana inicial.

Vaig plantajar el projecte d'aquesta manera, amb objectius cada setmana, i anant a visitar les empreses abans de dissenyar el prototip per estalviar-me feina si no el veien viable, i també un cop finalitzat tot els desenvolupament per veure com s'havia plasmant l'idea que els hi havia comentat en el paper, a un projecte físic i funcional.

### 4.2 Planificació final amb diagrama de Gantt

Com s'ha explicat anteriorment en l'apartat anterior, a causa de la pandèmia, i en sortir un imprevist, la idea del supermercat no va assolir els objectius que em vaig proposar, va girar-me tota la planificació inicial, finalment vaig anar fent sobre la marxa, sempre dins una organització.

La línia temporal del projecte, Objectiu/setmana no ha estat organitzada com tenia previst, però he anat assolint els objectius setmana a setmana per atrapar la feina i no quedar-me curt de temps, si que me n'ha faltat més temps en la documentació, per explicar millor les coses.

El diagrama seria una cosa així, amb petites variacions, ja que no ha estat planificat amb antelació, sinó a posteriori per comprovar el recorregut fet durant el projecte.

Activitat/ Setmana	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Pensar idea	✓	✓				✓	✓										
Contactar/Visitar Empreses		✓	✓														
Dememar Material Prototip			✓	✓													
Muntatge del prototip					✓	✓											
Programació APP								✓	✓	✓	✓	✓	✓	✓	✓		
Debug/Arreglar errors															✓	✓	
Documentació					✓	✓								✓	✓	✓	
Preperar Presentació																	✓

Figura 5: Diagrama Gantt d'objectius/setmana final.

Vaig desenvolupar el primer prototip molt ràpid, i ja vaig veure que la cosa no funcionava com esperava, ràpid vaig posar-me a pensar en una altra idea. Finalment vaig estar moltes setmanes barallant-me fent l'aplicació perquè el llenguatge d'Android no l'havia tocat, i se m'ha fet un pèl difícil l'autoaprenentatge.

## 5 Marc del treball i conceptes previs

El marc del treball el tenia molt ben pensat, sabia que volia anar de cara a un mercat logística o un supermercat de primeres, que és on el meu projecte agafa més força. A les empreses on va adreçat la meva idea són clients en potència, ja que la tecnologia en aquests sectors fa anys que no es reinventa, un bon projecte que els hi aportés tecnologia nova i facilitats per fer les diferents tasques amb més rapidesa, obtindria un potencial enorme per convertir-se en un negoci, venguent el nostre prototip.

Per entendre millor les idees desenvolupades la idea inicial era aconseguir que en un supermercat no s'hagi de tenir una persona treballant a cada caixa i anar passant cada producte amb el seu codi de barres, el que volia aconseguir és cada producte amb el seu xip RFID i agilitzar el procés de pagament, de tal manera que el client passant el carro pel sensor ja es generés la llista de tota la compra. La idea la vaig implementar amb un Arduino i un sensor RFID, de tal manera que passant els productes s'anessin registrant amb més rapidesa que el codi de barres, i sense tenir visible el xip.

Aquesta idea no va agafar gaire força a mesura que l'estava desenvolupant, (Totes les decisions es detallen a la secció 7) així que vaig optar per alguna cosa més pràctic i és utilitzar el telèfon mòbil per agilitzar els processos d'entrada i sortida de mercaderies, també en funció d'organització en el magatzem, ja que les visites fetes a empreses de Girona moltes portaven una mala gestió de magatzem, i no sabien on tenien els paquets molts cops.

L'aplicació que he dissenyat consta d'unes vistes del magatzem, per tal de llegir els xips NFC i registrar-los a la posició assignada, un cop busquis el paquet t'indicarà a quina posició del magatzem es troba i el podràs treure de la BBDD, és a dir que s'envia el paquet.

---

Un cop explicada tota l'evolució en grans trets del meu projecte, i les modificacions que han sorgit, en els següents apartats explicaré més detalladament totes les decisions, i l'aplicació final.

## 6 Requisits del sistema

### 6.1 Requisits de l'aplicació Warehouse-Sync

En aquest apartat parlarem de l'aplicació "Warehouse sync", nom que l'hi he posat en aquesta aplicació, cal definir-la com una aplicació de mòbil capaç de gestionar l'entrada i la sortida dels paquets del magatzem, a partir de xips NFC. Amb una funcionalitat afegida, és capaç de localitzar-nos el paquet dins del mapa, per saber el passadís, l'estanteria i el prestatge on es troba, facilitant la feina a l'operari del magatzem per trobar el paquet.

L'aplicació funcionarà al cim d'Android, sistema operatiu en qualsevol mòbil estàndard, també necessita que tingui lector de NFC. (Avui en dia pràcticament tots en tenen)

Estarà programat amb el IDE "Android Studio", seguidament explicaré les funcionalitats que hauria de tenir l'aplicació per tal d'aconseguir un prototip per mostrar els clients.

Els requisits de l'aplicació són els següents:

#### Requeriments i funcionalitats de Warehouse sync

1. Tenir unes credencials per seguretat a l'hora d'entrar-hi. (Usuari i Contrasenya)
2. Tenir un menú on podem veure les diferents opcions de l'aplicació.
3. Visualitzar tot el magatzem, amb les estanteries i passadissos.
4. Poder prémer una estanteria del magatzem per registrar-hi un paquet.
5. Llegir el xip NFC del paquet.
6. Poder registrar el paquet en un lloc del magatzem, amb el codi NFC, Adreça i nom del paquet.
7. Poder buscar i seleccionar un paquet ja registrat.
8. Visualitzar el paquet en el mapa del magatzem.
9. Esborrar paquets del magatzem.
10. Sortir de l'aplicació.
11. Que l'aplicació sigui funcional i senzilla d'utilitzar

---

## 7 Estudis i decisions

### 7.1 Primera Idea: Supermartek

#### 7.1.1 Explicació i evolució de l'idea

La idea del meu projecte era aconseguir que en un supermercat no s'hagi de tenir una persona treballant a cada caixa i anar passant cada producte amb el seu codi de barres, el que volia dur a terme des d'un principi, era modernitzar el codi de barres i que cada producte portés el seu xip RFID fet que faria agilitzar el procés de pagament, de tal manera que el client passant el carro amb els productes pel sensor de xips NFCs, ja es generaria la llista de tota la compra. D'aquesta manera podríem evitar infinites cues als supermercats.

Donant voltes a la meua idea no com informàtic, sinó més en punt de vista de Marketing, quedava molt antiestètic vendre la idea com a "substitució dels operaris del supermercat per sensors", també enfocant que el supermercat era el client final donant-li avantatges a ells, ja que són els compradors, vaig veure convenient canviar la manera d'enfocar el projecte fent petits canvis més en l'àmbit comercial i funcional, però no pel que fa al dispositiu (Hardware+Software).

El replantejament de la idea vaig aconseguir dut a terme una lleugera reforma, no en el model funcional sinó en la finalitat per què es farà servir. Com s'ha descrit abans cada producte tindrà el seu xip RFID, però el sensor no es farà servir en la caixa, en aquesta evolució que va patir el projecte, el sensor s'utilitzarà en el magatzem del supermercat, un cop arribi el camió del proveïdor passaran el palet pel sensor, així determinant quants productes han entrat i guardant-los a la base de dades, agilitzant molt el procés de magatzem d'un supermercat, i de cara a un futur altres empreses de logística. Estalviant que cada paquet/palet s'hagués de llegir el codi de barres, i registrant-lo com a estoc del magatzem. També facilitaria molt en l'àmbit de fer inventari per diferents empreses on tinguessin magatzem, simplement podrien passar el dispositiu pels xips NFC, fent un recompte automàtic i molt més ràpid que el tradicional.



Figura 6: Logo del primer projecte Supermartek.

Un cop amb l'idea pensada i ja mig desenvolupada, havia escollit hardware barat, ja que per competir contra les etiquetes del codi de barres, que són molt barates era imprescindible que el hardware éssers barat. Però pel tema de pressupost el dispositiu que vaig desenvolupar no complia els objectius més importants que volia assolir de cara a un bon funcionament del projecte.

En resum el sensor només llegia els xips un per un, i no a la distància que hauria d'haver llegit amb la tecnologia NFC, que serien uns 5-20cm on el meu dispositiu només llegia si el xip estava enganxat al sensor, si hagués agafat xips NFCs actius i sensor més bons segurament s'hagués assolit perfectament els objectius, però l'idea era competir contra la tecnologia actual, els codis de barres que són molt barats.

Tot i no assolir els objectius no vaig donar el projecte com a defallit, sinó com a possible opció de millora perquè el meu projecte era d'investigar i dur a terme una idea, era el moment de pensar en un

---

altre dispositiu per portar al mercat els avantatges dels xips NFCs.

Un cop amb la desil·lusió del projecte, havia d'aconseguir noves metes i noves idees que pogués implementar, així que la millor manera era anar a dins del sector de la logística/supermercats, per veure la manera de treballar de les diferents empreses del sector, fer preguntes al respecte sobre el meu projecte, i aconseguir una idea beneficiosa per poder implementar i ser possible producte a vendre en aquest mercat.

Quan vaig contactar amb diferents empreses locals de Girona, com eren Nacex, Dachser, Maxi Dia, entre d'altres, era el Febrer principis de Març, i per culpa de la pandèmia van cancel·lar-se forces visites que tenia previstes a empreses, vaig aconseguir que confirmessin visita a 4 empreses. Els hi vaig portar diferents preguntes a fer, per exemple si era viable la meua primera idea del "Supermartech", a l'hora de fer inventari, com treballaven, com portaven la gestió d'entrada de mercaderies, veure una mica el magatzem i la manera de treballar que tenien, etc...

Vaig aclarir moltes coses que tenia al cap desorganitzades amb les diferents visites, el supermartech que els hi portava no la veien gaire profitosa, ja que no complia els objectius que m'havia proposat, veien que comprar un altre dispositiu que permetés llegir els xips NFC era com "estúpid", si el codi de barres porta temps en el mercat, és barat i compleix els resultats esperats, l'idea que em van donar diferents dels encarregats va ser que si podia portar els xips NFC a algo més portàtil com per exemple un mòbil, sí que la veurien més viable, ja que molts dels operaris del magatzem, camioners i transportistes tenien mòbils propis de l'empresa, i seria un camí més viable que dissenyant un sensor NFC extern.

Veient la manera de gestionar d'alguna de les empreses els seus paquets, era força desastrós, tenien molts palets i paquets literalment tirats pel mig del magatzem, i quan n'havien de buscar un havien de buscar durant uns minuts fins a trobar-lo, amb la idea del mòbil i veient la mala gestió per part d'algunes de les empreses visitades, vaig estar pensant durant uns dies alguna idea on implementar les diferents coses en una.

Finalment vaig donar amb una idea viable, que solucionés tots els problemes vistos en les visites al sector de la logística, que es pogués implementar de manera senzilla i utilitzar des del telèfon mòbil, seria molt barata, ja que l'únic que entre en joc serien els xips NFC, i podria portar els avantatges descrites en capítols anteriors dels xips NFC respecte a els codis de barres.

L'idea va ser en dissenyar un prototip d'una aplicació per dispositius mòbils amb el SO Android (les empreses no compraven mòbils Apple pels treballadors), capaç de gestionar la base de dades amb l'entrada dels nous paquets, portar una localització interactiva d'on es troben els paquets dins del magatzem de manera ràpida, útil i molt senzilla d'utilitzar. Va sorgir la idea del Warehouse-sync.

El disseny del logo i de la icona van anar referits als paquets que permet gestionar l'aplicació, també volia transmetre un to Blau en els menús i disseny de les diferents vistes per donar confiança i professionalitat al client.





(a) Logo warehouse-sync



(b) Icona de l'aplicació

Un cop amb el disseny i la idea de l'aplicació vaig començar a dissenyar els diferents requeriments del prototip que permetés veure les principals funcionalitats en els clients. Així que decidint els requeriments vaig posar coses òbvies com registrar entrada de paquets, eliminar paquets, buscar i veure tots els paquets guardats a la base de dades, i com s'havia dit anteriorment portar una localització molt senzilla i visual per saber on està cada paquet registrat en la base de dades, per oferir una facilitat als operaris de trobar el paquet en segons.

*Cal deixar constància que és un prototip amb les funcionalitats més bàsiques i importants, en el producte final de cara a un futur podran haver-hi millores i noves funcionalitats convenients per anar millorant l'aplicació i que sigui el màxim complet possible.*

Totes les empreses actuals ja utilitzen bases de dades, així que vaig veure convenient incorporar un mínim de base de dades per mostrar el client que es pot adaptar en les seves BBDD sense cap problema, per gestionar i crear la base de dades de l'aplicació he utilitzat Room, que és una capa que ens ajuda a l'hora de sol·licitar processos i gestiona una base de dades SQLite, més ben explicat en capítols sobre la implementació de l'aplicació.

Pensant en la seguretat de l'app, vaig veure convenient posar-hi una pantalla d'identificació amb usuari i contrasenya, així l'operari cada dia que vagi al torn de treball es pot connectar sense cap problema, la contrasenya es podria canviar a cada torn de treball, així evitem que algun treballador que ja ha plegat accedeixi a l'aplicació, molt senzill de canviar la contrasenya amb la BBDD.

L'altra decisió que vaig prendre sobre el tema d'identificació és que no es pugui registrar, ja que com empresa nosaltres portarem les vendes de l'aplicació, i ens facilitarà més el tema de llicències, i qui pot o no utilitzar l'aplicació, evitant que persones puguin accedir-hi i utilitzar-la sense haver-la comprat.

El model de negoci que portarem a terme a l'hora de vendre l'aplicació, és vendre llicències i tot el procés d'adaptar l'aplicació en la Base de dades del client, possibles implementacions que desitgin a més a més de les funcionalitats que portarà l'aplicació, configuracions de les opcions de l'aplicació com poden ser el mapa del magatzem, que s'hauria de configurar per cada client perquè no tots els magatzems són iguals, la informació que guarden dels paquets, etc.

En el mapa del magatzem volia utilitzar el tipus d'imatge SVG, que són gràfics vectorials redimensionables, ens permeten escalar les imatges de manera que no és pixelin i dissenyar les imatges de manera molt senzilla, però per fer el prototip, ja que havia implementat 2 idees vaig escollir la manera conven-

---

cional, de fer passadissos, estanteries i prestatges dinàmics, podria ser una possible millora per futures versions de l'aplicació, ja que per temps no he pogut implementar-ho, seria molt útil posar una lupa per fer gran i petita tots els passadissos del magatzem per empreses grosses. També ens ajudaria molt a l'hora de configurar els diferents magatzems de tots els clients, ja que el sistema SVG utilitza fitxers XML per crear l'imatge.

A la vista dels prestatges, es genera amb un recycleview, ens evita haver de guardar layouts per cada estanteria de cada passadís, simplement quan es crida l'activitat dels prestatges el recycleview es genera de nou, important la informació de la base de dades, en cas que contingui un paquet algun dels prestatges. Aquesta decisió la vaig prendre, a posteriori perquè no estava acostumat a utilitzar els recycleviews, intentant fer-ho del mètode convencional, al cap d'uns dies vaig veure que guardar layouts per cada estanteria era massa costós.

Aquests són les principals decisions que he anat encaminant a mesura que feia el projecte i sorgien problemes o dissenys a fer, no obstant segur que hi ha petites coses que no he tingut en compte a l'hora d'explicar les decisions, perquè eren detalls petits, però de tots els camins seguits ha sortit un projecte amb cara i ulls i un producte molt interessant.

---

## 8 Anàlisi i disseny del sistema

En aquest apartat bàsicament explicarem com funciona l'aplicació, els conceptes bàsics de totes les funcionalitats i menús que comporta el nostre software.

S'explicarà l'esbós de cada pantalla que conté WareHouse-sync el funcionament que havien de tenir cada pantalla, i com es poden executar les diferents opcions del menú de forma senzilla.

### 8.1 Cas d'ús de WareHouse-sync

Amb els requeriments pensants, abans de disposar-me a pensar el disseny de les pantalles, vaig fer un diagrama de cas d'ús general per plasmar les idees sobre l'aplicació i tenir el disseny més organitzat. Això em va facilitar molt la feina, perquè la feina més dura era aprendre a programar amb android, com més ben planificat tot facilitaria molt el desenvolupament a posteriori.

L'esquema de cas d'ús general:

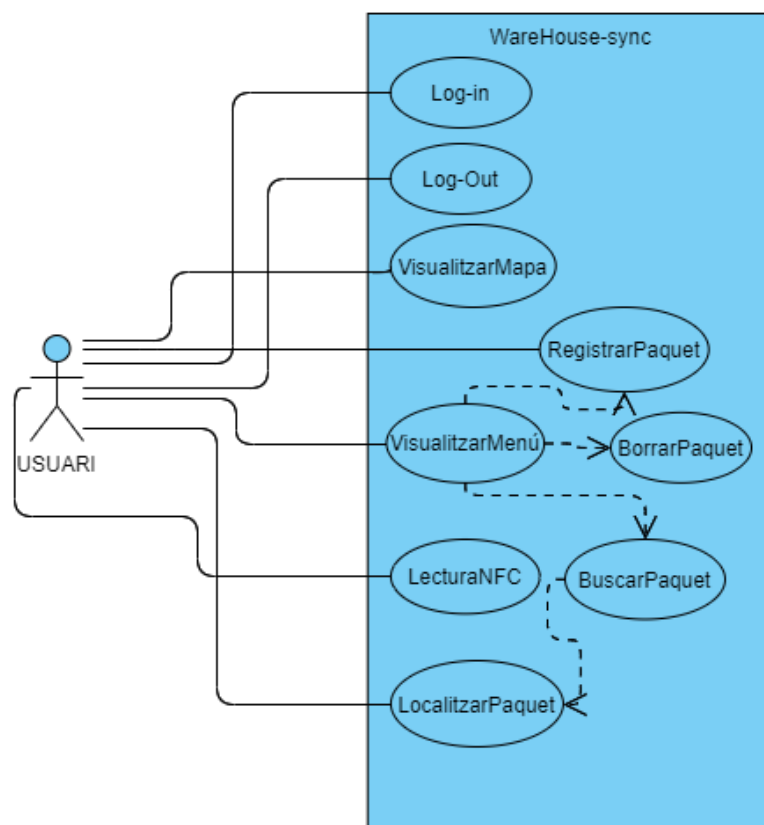


Figura 7: Diagrama cas d'ús general

---

## 8.2 Casos d'ús individuals

Explicarem tots els casos individuals en forma de taula, explicats per tenir el projecte més detallat, per descriure'ls s'utilitzarà l'estructura següent:

Títol del cas d'ús	
Cas d'ús:	
Actor:	
Objectiu:	
Precondicions:	
Escenari:	
Postcondicions:	
Excepció:	

1. **Títol del cas d'ús:** Nom del cas d'ús per identificar-lo.
2. **Cas d'ús:** Descripció breu del cas d'ús.
3. **Actor:** Agent que interectua amb el cas d'ús.
4. **Objectiu:** Descripció detallada del cas d'ús.
5. **Precondicions:** Condicions inicials que s'han de complir per realitzar-se.
6. **Escenari:** Conjunt de passos que s'ha de donar per realitzar el cas.
7. **Postcondicions:** Estat del sistema despres de realitzar-se.
8. **Excepció:** Incidències o alternatives que pugui haver al executar-se.

Log-in	
Cas d'ús:	Autenticació de l'usuari
Actor:	Usuari de l'aplicació
Objectiu:	Comprovar que s'han introduït correctament les credencials
Precondicions:	Tenir la conta registrada, i saber la contrasenya
Escenari:	<ol style="list-style-type: none"><li>1. S'introdueix l'usuari</li><li>2. S'introdueix la contrasenya</li><li>3. S'inicia sessió</li></ol>
Postcondicions:	usuari connectat
Excepció:	En cas d'entrada incorrecte de credencials surt error i repeteix el procés

Visualitzar magatzem	
Cas d'ús:	L'usuari visualitza el magatzem
Actor:	Usuari de l'aplicació
Objectiu:	L'usuari pot moure's fent scroll per la vista del magatzem, pot veure el botó de menú i també pot accedir als prestatges polsant una estanteria.
Precondicions:	L'usuari pot entrar de 3 maneres diferents, normal, selecció o localització
Escenari:	<ol style="list-style-type: none"><li>1. <b>Normal:</b> Moure amb scroll pel magatzem, polsar menú i polsar estanteria.</li><li>2. <b>Selecció</b> Polsar estanteria</li><li>3. <b>Localització</b> Polsar estanteria pintada</li></ol>
Postcondicions:	Accedeix al menú si pulsa menú, accedeix a la vista de prestatge si pulsar una estanteria.
Excepció:	FALSE

Accedir Menú	
Cas d'ús:	L'usuari visualitza el menú
Actor:	Usuari de l'aplicació
Objectiu:	L'usuari pulsa el botó de menú, i li permet veure i seleccionar les diferents opcions
Precondicions:	L'usuari ha d'estar en la vista del mapa del magatzem
Escenari:	Usuari pulsa el botó del menú
Postcondicions:	Desplega el menú visualitzant les diferents opcions
Excepció:	FALSE

Log-Out	
Cas d'ús:	L'usuari es desconnecta de l'aplicació
Actor:	Usuari de l'aplicació
Objectiu:	L'usuari es desconnecta de l'aplicació, deixant de poder utilitzar-la, fins al pròxim log-in
Precondicions:	L'usuari ha d'estar en el menú
Escenari:	Usuari pulsa el botó log-out del menú
Postcondicions:	DUsuari desconnectat, layout de log-in
Excepció:	FALSE

Registrar paquet	
Cas d'ús:	L'usuari pot registrar un paquet
Actor:	Usuari de l'aplicació
Objectiu:	L'usuari introdueix totes les dades del paquet, l'objecte és registrat a la base de dades
Precondicions:	Totes les dades del paquet han de ser introduïdes
Escenari:	<ol style="list-style-type: none"> <li>1. Usuari entra el nom del paquet</li> <li>2. Usuari llegeix el xip NFC</li> <li>3. Usuari introdueix l'adreça</li> <li>4. En cas de localització no seleccionada, pulsar el botó seleccionar</li> <li>4. Usuari pulsa el botó per registrar el paquet a la base de dades</li> </ol>
Postcondicions:	El paquet serà guardat a la base de dades
Excepció:	Alguna de les dades no ha estat introduïda sortirà un missatge d'error que falten dades

Buscar paquet	
Cas d'ús:	L'usuari pot buscar un paquet
Actor:	Usuari de l'aplicació
Objectiu:	L'usuari visualitza tots els paquets registrats fins a la data, i pot buscar-ne un de concret
Precondicions:	Nombre de paquets registrats a la base de dades != NULL
Escenari:	<ol style="list-style-type: none"> <li>1. Entra el nom del paquet</li> <li>2. L'usuari selecciona una per localitzar-lo</li> </ol>
Postcondicions:	En cas de paquet seleccionat, activitat de localització llançada
Excepció:	FALSE

---

Esborrar paquet	
Cas d'ús:	L'usuari pot esborrar un paquet
Actor:	Usuari de l'aplicació
Objectiu:	L'usuari visualitza tots els paquets registrats fins a la data, i pot buscar-ne un de concret pel nom, o llegint el NFC, si l'esborra no es podrà tirar enrere
Precondicions:	Nombre de paquets registrats a la base de dades != NULL
Escenari:	<ol style="list-style-type: none"> <li>1. Entra el nom del paquet</li> <li>1.1 O llegeix el codi NFC</li> <li>2. L'usuari selecciona un paquet</li> <li>3. L'usuari pulsa esborrar</li> </ol>
Postcondicions:	Paquet esborrat de la base de dades
Excepció:	FALSE

Localitzar paquet	
Cas d'ús:	L'usuari veu el paquet pintat en el mapa
Actor:	Usuari de l'aplicació
Objectiu:	L'usuari visualitza el mapa del magatzem amb l'estanteria que conté el paquet pintat, només pot accedir en aquella estanteria per veure a quin prestatge es troba
Precondicions:	Seleccionar un paquet en l'activitat Buscar paquet
Escenari:	1. L'usuari pulsa el botó pintat verd de l'estanteria
Postcondicions:	Llista de prestatges amb el que conté el paquet pintat
Excepció:	FALSE

---

### 8.3 Esbós de les pantalles de l'aplicació

Un cop amb la idea de tots els requeriments i funcionalitats que havia de tenir l'aplicació, vaig fer uns esbossos de cada pantalla, per seguir una pauta a l'hora de programar-la, a continuació explicarem una mica cada pantalla les funcionalitats que hauria de tenir seguint els requeriments.

La principal idea era que fos molt fàcil d'utilitzar per agilitzar tot el procés, així que veurem el primer disseny senzill de les diferents pantalles del prototip de l'aplicació.

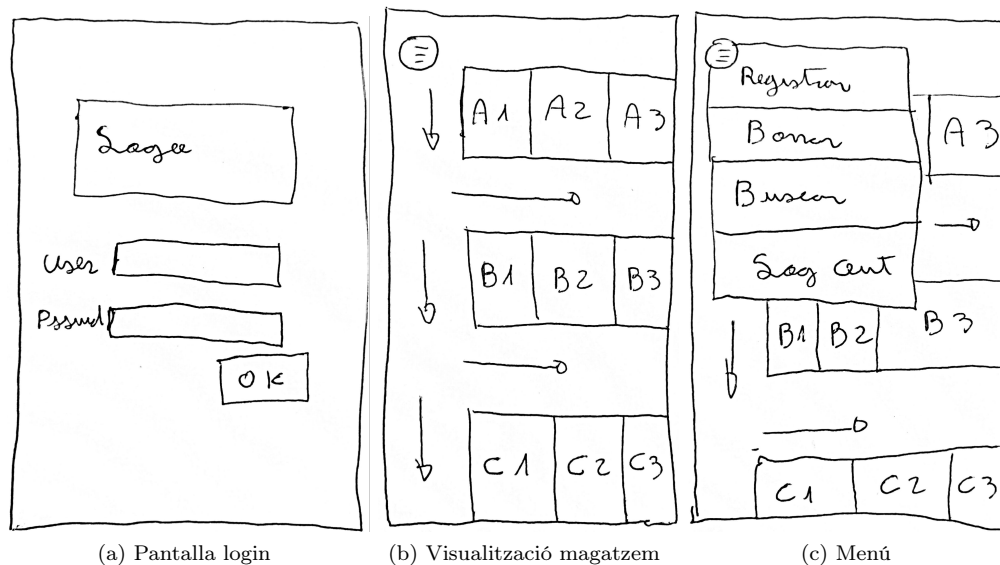


Figura 8: Esbós de les pantalles de l'aplicació

- a) ) La primera pantalla en la Figura 8a i la més essencial per la seguretat de l'aplicació, és la pantalla de Log-in, ens demanarà l'usuari i la contrasenya de l'empresa per tal d'entrar al seu perfil i utilitzar totes les funcions de l'aplicació amb el magatzem d'aquella empresa.
- b) ) La Figura 21 ens mostra el que serà la pantalla principal de l'aplicació, on podrem veure tots els carrils i estanteries del magatzem, i desplaçar-nos per ella, si en seleccionem una es podrà registrar un paquet en aquella posició.
- c) ) La tercera Figura 21 podem veure el menú, on es podrà registrar, eliminar, buscar un paquet i sortir de l'aplicació.

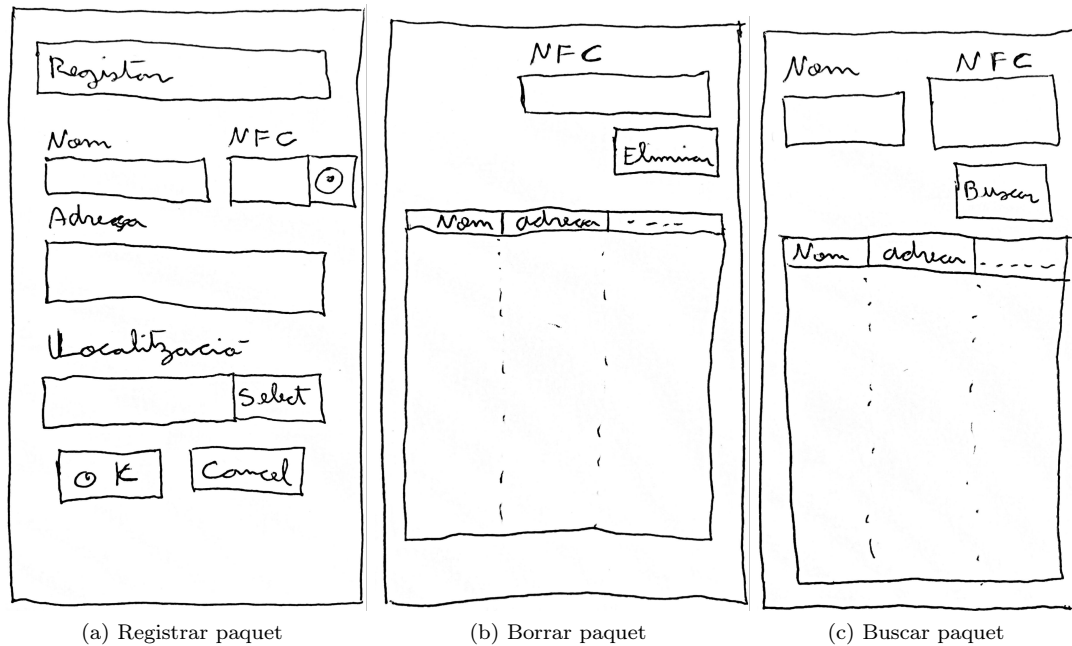


Figura 9: Esbós de les pantalles de l'aplicació 2

- a) ) La primera pantalla en la Figura 9a ens demana els diferents camps per registrar un paquet, en aquest prototip posarem Nom, Adreça, ubicació dins del magatzem i el Codi NFC que tindrà un botó per llegir-lo.
- b) ) La Figura 23b ens permetrà escollir un paquet de la llista, o buscar-lo amb el seu NFC per tal de borrar-lo del magatzem, *exemple: Quan el paquet surt del magatzem i passa al transportista.*
- c) ) La tercera Figura 9c ens permet veure tots els paquets registrats a la BBDD, i un cop en seleccionem 1, es mostrarà la seva posició en el magatzem.

Això és un esbós que vaig dissenyar a l'hora de programar l'APP, cal dir que hi pot haver algun canvi en el format final, amb aquest disseny s'ha buscat la simplicitat a l'hora de registrar paquets i buscar-los dins del magatzem, per potenciar les característiques dels xips NFC.



---

## 9 Implementació i proves

### 9.1 Primera Idea: Supermartech

Com s'ha explicat en apartats anteriors, aquest prototip consistia en un dispositiu capaç de llegir diferents productes cada un amb el seu codi NFC, barrejats en un carro de supermercat i aconseguís llegir-los. Procedirem en explicar tot el procés del disseny, i els objectius que malauradament no s'han complert com s'esperava.

#### 9.1.1 Disseny i implamentació del prototip

##### 9.1.1.1 Material necessari

Per fer el prototip del Supermartech i veure totes les funcionalitats esmentades, es necessitarà el següent material, sempre buscant competir amb el codi de barres així que els preus han de ser baixos i no podem accedir-nos, material utilitzat:

##### Llista material pel prototip

1. Arduino UNO - 10
2. RFID-RC522 - 9
3. Chips RFID - 2 (8Uds)
4. Components generals del muntatge(Resistències, leds, cables...) - 4

**Total:** 24

Com a software s'ha utilitzat la IDE de l'arduino, que és gratuït, i també com a base de dades per fer les proves a l'Excel s'ha utilitzat un plugin anomenat PLX-DAQ. Ens permet connectar l'excel amb l'Arduino, i enviar informació entre els dos, molt fàcil d'utilitzar per simular una base de dades que podria tenir el client.

Al ser la primera idea, vaig anar a el que és bàsic, per tal de fer un muntatge ràpid i veure el seu funcionament, si complia els objectius o no, si era viable, etc. Podria tenir moltes millores de components, però el pressupost havia de ser econòmic.

### 9.1.1.2 Connexió de components

Les connexions del projecte, de tots els components necessaris amb l'arduino els podem veure de manera visual en figura 10 són molt senzills, en total s'utilitzen 10 pins digitals i no s'han utilitzat pins analògics.

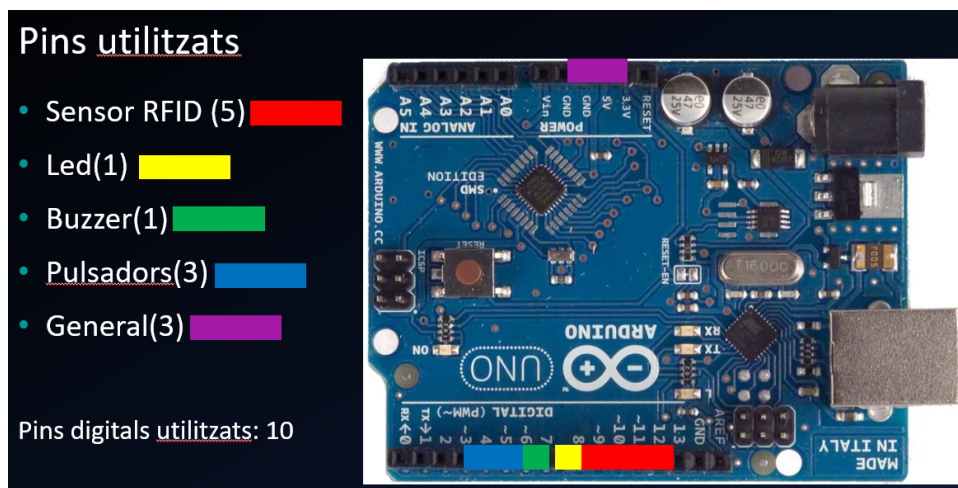


Figura 10: Pins arduino UNO utilitzats.

M'agradaria incloure l'esquema de les connexions que s'ha utilitzat per al xip RFID, ja que és el més complicat de tots. Les altres components són molt senzilles, surt el cable del pin de l'Arduino fins a la component (*una resistència en mig si és necessària*), i de la component fins al terra. En canvi el xips RFID-RC522 té diferents pins que s'han de connectar, així que les connexions queden igual que l'esquema següent:

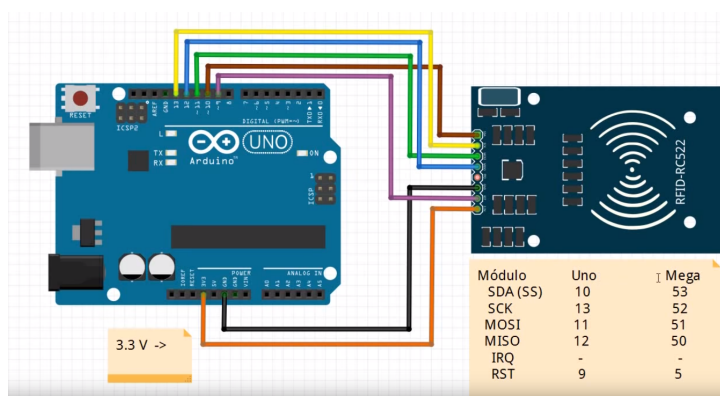


Figura 11: Esquema elèctric del chip RFID.

Com es pot veure en la figura 12 tenim tot el muntatge de la primera idea Supermartech, els objectius eren llegir els paquets d'entrada de mercaderies sense haver de buscar el codi de barres de cada paquet, simplement acostant el sensor en els diferents xips NFCs inclosos en els paquets/palets.

---

La presentació del prototip no està ben acabada, era una prova per tal de determinar la viabilitat de la idea i si complia els objectius esperats.

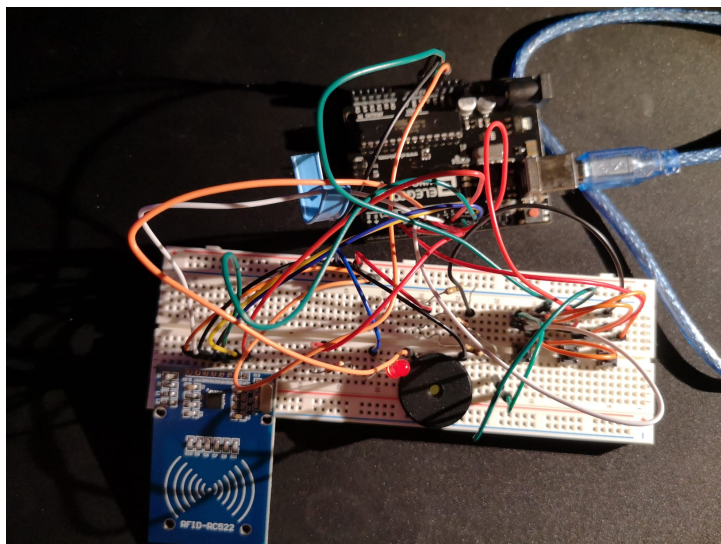


Figura 12: Prototip inicial supermartech

### 9.1.1.3 Programació de l'arduino

El programa dissenyat és una màquina d'estats, amb 5 estats. S'ha escollit fer una màquina d'estats, perquè a la hora de controlar un programa quan es va repetint en loop infinit és la millor manera de controlar totes les entrades i sortides de manera fàcil.

#### Estructura del programa

1. Estat 0  
Inicialitza totes les variables  
*Estat de reset*
2. Estat 1  
Espera el codi del producte
3. Estat 5  
Llegeix tots els productes amb aquell codi(chips RFID)
4. Estat 10  
Transferència de dades
5. Estat 11  
Comprova flag dels botons i torna a l'estat 0

**Aclariment:** Com es pot veure els estats són saltats, no consecutius, és simplement que m'agrada programar deixant de marge 5 nombres entremig de cada estat, perquè sempre t'acabes deixant alguna cosa i és més senzill afegir un estat.

---

#### 9.1.1.4 Configurar PLX

Un cop el prototip està connectat a la corrent, s'ha d'obrir la base de dades, en el nostre cas l'Excel amb el plugin PLX. Es poden configurar varies coses més sofisticades, però en el nostre cas només necessitarem dir-li el port on tenim connectada l'Arduino en el nostre PC i amb quina freqüència es comuniquen, en aquest cas 9600. Un cop configurat li donem al botó de connect, i l'Arduino ja estarà preparada per enviar la diferent informació.

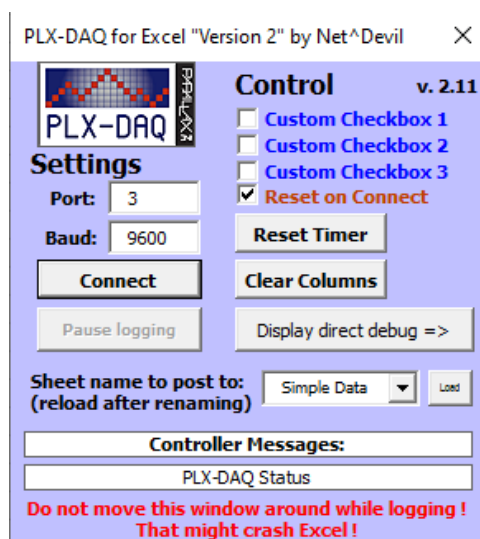


Figura 13: Pantalla PLX configuració.

#### 9.1.1.5 Entrar codi producte

Quan l'Arduino ja està connectada a la nostra base de dades, ja podem començar a utilitzar-la. L'Arduino estarà a l'estat d'esperar el codi del producte, per dissenyar el prototip inicial s'han inclòs 3 botons per tal d'accionar aquest procediment de la manera menys costosa.

Tenim 3 botons, els 2 primers anomenats **B1** i **B2** en la figura 15, són per donar el codi del producte, al ser 2 botons s'ha utilitzat un codi binari, si un botó és polsat, ja obté el valor de 1, en cas de no ser polsat és 0.

De tal manera que obtenim diferents opcions de codis 00, 01, 10, 11. Quan s'ha entrat el codi de producte, tenim el 3r botó, anomenat **OK**, que és per confirmar i passar al següent estat de llegir productes.

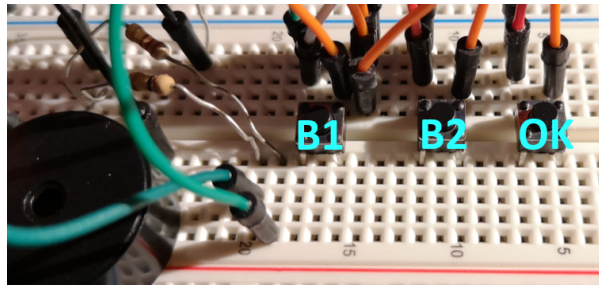


Figura 14: Botonera del prototip.

#### 9.1.1.6 Entrar chips rfid dels productes

Ja només queda passar tots els xips RFID referents al nostre codi de producte pel sensor, *el sensor del prototip no agafa a més de 2-4cm*. Un cop quan ja s'han acabat tots els productes d'aquella marca/tipus, es torna a pulsar el botó de confirmació **OK**.

L'Arduino enviarà missatges dels nostres productes entrats al magatzem en la base de dades, com la data d'entrada, el tipus de producte i el codi RFID registrat per aquell producte.

Següent pas seria entrar un altre codi de producte i repetir tots els passos.

#### 9.1.2 Objectius

Crec que no s'han aconseguit els objectius principals. S'ha dissenyat un prototip inicial, que té totes les funcionalitats de la idea del projecte, també s'ha inserit la comunicació via Excel, simulant una base de dades simple.

En tots els projectes d'investigació es troben mancances en la idea fonamental i/o problemes, en el meu cas, s'han trobat limitacions de hardware en la tecnologia RFID. Per tant, aquests objectius que m'hagués agradat complir, ja que eren els innovadors respecte a els processos duts a terme dintre dels magatzems, s'han inclòs en la llista de no aconseguits.

Al ser els objectius que podien compatir contra el codi de barres, per agilitzar processos i millorar l'entrada d'estoc en els magatzems, passant el palet enter pel sensor, i detectant tota la informació de cada paquet. Com que no he aconseguit llegir més de 4CM i només llegir 1 xip a l'hora ens limita molt, així que dono aquest prototip com a defallit pel hardware.

Però cal dir que està sortint una tecnologia nova, l'ultra-wideband, que promet llegir a una distància de 20cm-200m, i varis xips a l'hora. S'haurà d'observar la tecnologia de prop per tal d'actualitzar el nostre hardware i fer un prototip que pugui entrar dins del mercat. [8]

---

## 9.2 Segona Idea: WareHouse-Sync

### 9.2.1 Explicació estructura del projecte Android Studio

L'estructura en si és la mateixa que podem trobar en qualsevol aplicació programada amb Android Studio. Té les diferents layouts amb XML, les activitats de cada cas d'ús, la base de dades que explicarem a continuació, i les diferents classes d'objectes que tenim i que guardem a la base de dades.

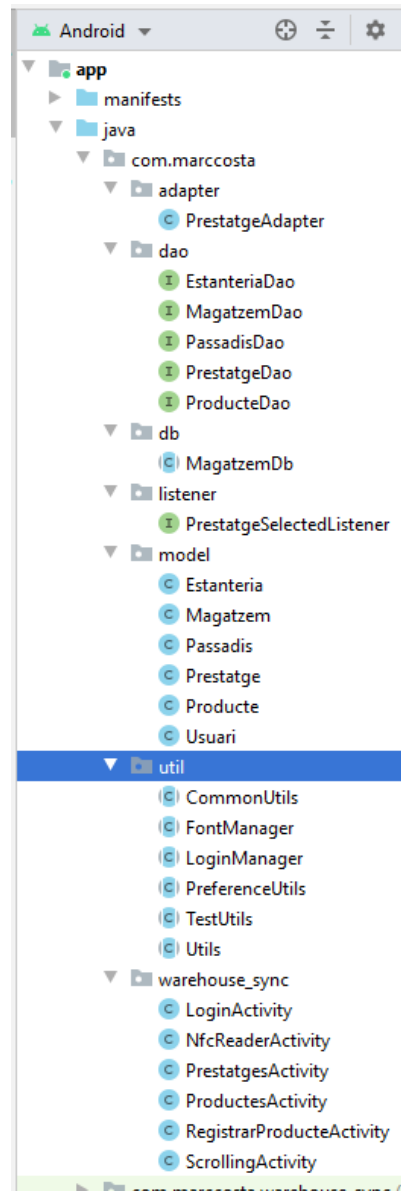


Figura 15: Estructura projecte Android Studio.

---

### 9.2.2 La base de Dades en local (ROOM)

Per crear la base de dades que gestiona tota la informació de l'aplicació, he utilitzat Room. És una llibreria molt fàcil d'utilitzar, que ens permet gestionar bases de dades SQLite en Android de manera segura.

SQLite és un sistema de domini públic que gestiona bases de dades relacionals, i l'avantatge que ens proporciona és que no funciona amb un procés independent, sinó que forma part de l'aplicació en si. Per tant no es necessita un servidor per llançar-li peticions a la Base de Dades ni que res sigui instal·lat a part de l'aplicació, guarda permanentment una base de dades local en el dispositiu d'una manera força òptima.

Room és una llibreria que implementa una capa intermitja entre la base de dades SQLite i la nostra aplicació, ens facilita la feina en cas que s'hagués de canviar la base de dades a la llarga, sinó utilitzéssim Room hauríem de canviar moltes injeccions SQL de la nostra aplicació.

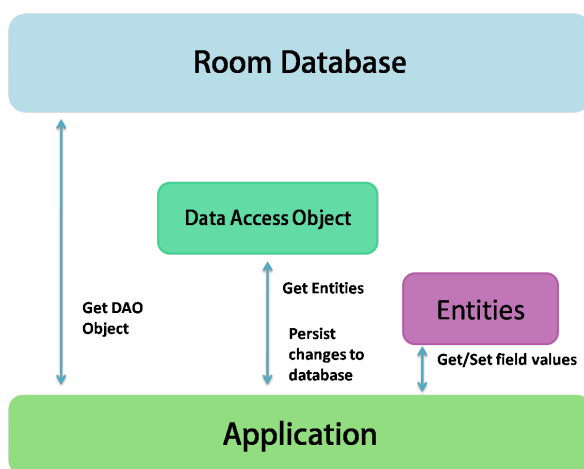


Figura 16: Esquema de crides de Room i l'aplicació.

L'esquema d'aquesta base de dades és força simple, ja que no deixa de ser un prototip de cara a un producte final amb més funcionalitats, i amb una BBDD d'una empresa real. Com es pot veure en la següent figura:

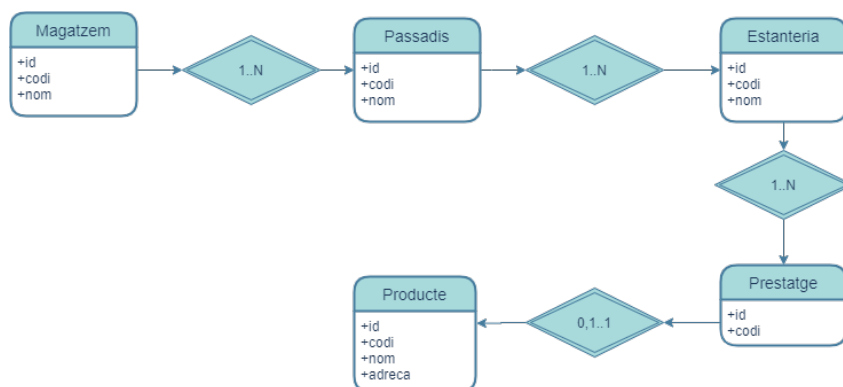


Figura 17: Esquema de la BBDD de l'aplicació.

---

Com podem veure tenim 1 magatzem guardat en la base de dades que és la classe principal de l'aplicació. El magatzem te 1 o molts passadissos amb la nomenclatura de la A-Z.

Cada passadís tindrà les seves estanteries anomenades per número, així ens facilita determinar les posicions del magatzem amb B3, sabem que és el passadís B l'estanteria 3.

Cada estanteria tindrà els seus prestatges també codificats amb un nombre creixent i cada prestatge podrà tenir 1 o cap producte.

Aquesta base de dades és la que forma l'aplicació de WareHouse-Sync, permeten la localització de paquets, el seu registre i treure'ls del prestatge.

En cas de vendre l'aplicació a un client, s'hauria d'adaptar l'aplicació amb la seva base de dades pel correcte funcionament.

A cada classe que utilitzem en l'aplicació tenim la definició de la taula, els diferents camps que necessita la taula i les relacions vistes en l'esquema anterior. Veurem un exemple del fitxer "Passadis.java":

```
1 @Entity(tableName = "passadissos")
2 public class Passadis
3 {
4     @PrimaryKey(autoGenerate = true)
5     private int id;
6
7     @ColumnInfo(name = "codi")
8     private String codi;
9
10    @ColumnInfo(name = "nom")
11    private String nom;
12
13    @ForeignKey(entity = Magatzem.class, parentColumns = "id", childColumns = "
14        magatzemId", onDelete = CASCADE)
15    private int magatzemId;
16
17    @Ignore
18    List<Estanteria> estanteries;
19 }
```

Podem observar que creem l'entitat passadissos com una taula de la base de dades, i el camp id (integer) serà la clau primària, serà incremental cada passadís creat, ja que té l'opció autogenerate. Altra informació que seran columnes de la taula com el codi, i el nom del passadís. La definició de la relació 1-N amb Magatzem, relacionant el camp id de la taula "magatzem" amb el camp magatzemId de la taula passadissos.

La llista d'estanteries està ignorada com a informació de la taula, ja que també és relació 1-N, per tant la relació es crearà a la taula "estanteries", si la creéssim aquí també seria relació N\_N.

La creació de la base es pot trobar en el fitxer "MagatzemDb.java", com veiem la següent línia de codi creem la base de dades segons les nostres classes, magatzem, passadís, estanteria i prestatge. Cada fitxer de cada classe tindrà definits les relacions i les claus primàries, com s'ha vist en el punt anterior. Aquí simplement cridem una funció general que agafa totes les definicions de cada classe i construeix la base de dades seguin les relacions i els camps definits.

```
19 @Database(entities = {Magatzem.class, Passadis.class, Estanteria.class, Prestatge.
    class, Producte.class}, version = 2, exportSchema = false)
```



---

En aquest mateix fitxer també creem la funció que es connectarà i ens retornarà la instància de la base de dades creada, un cop obrim l'aplicació.

```
21     public static synchronized MagatzemDb getInstance(Context context) {
22         if (instance == null)
23             instance = Room.databaseBuilder(context.getApplicationContext(),
24                 MagatzemDb.class, DB_NAME)
25                 .fallbackToDestructiveMigration()
26                 .allowMainThreadQueries()
27                 .build();
28         return instance;
29     }
```

Finalment creem les funcions abstractes que ens retornarà els repositoris de les diferents classes que tenim, per utilitzar la interfície DAO, i cridar les diferents injeccions SQL de cada taula.

Exemple de creació de la funció abstracte:

```
30 public abstract MagatzemDao magatzemDao();
```

Els fitxers dao ens serveixen per obtenir les comandes SQL de cada taula, i gestionar la nostra base de dades de manera fàcil, tindrem les típiques funcions de delete, update, insert i també altres que es puguin necessitar en les funcionalitats de l'aplicació.

Podem veure un exemple de 2 funcions del fitxer "PrestatgeDao.java":

```
31     @Query("SELECT * FROM prestatges WHERE id = :id")
32     Prestatge getByPk(int id);
33
34     @Query("SELECT * FROM prestatges WHERE estanteriaId = :estanteriaId")
35     List<Prestatge> getAllByEstanteriaId(int estanteriaId);
36
```

En la primera crida obtenim l'objecte prestatge a partir del seu id. La 2a injecció SQL podem obtenir la llista de prestatges segons un id d'estanteria.

Tots els repositoris Dao de cada classe són molt semblants, així que no he vist necessari explicar més sobre la base de dades i com funciona Room.

### 9.2.3 Disseny de les activitats de l'aplicació

En aquest apartat es mostraran les implementacions d'activitats més importants, les funcions més rellevants de cada activitat, donant detalls del funcionament del codi mostrat, per tal d'entendre millor la manera que ha estat implementada l'aplicació i el seu funcionament en un punt de vista més tècnic com a programador.

#### 9.2.3.1 LoginActivity.java

Aquesta activitat és la primera a executar-se quan obrim l'aplicació mostrant la pantalla de login de la APP, i esperant que l'usuari entri correctament les credencials per entrar a utilitzar totes les funcions que permet Warehouse-sync.

```

37
38     protected void onCreate(Bundle savedInstanceState) {
39         super.onCreate(savedInstanceState);
40         setContentView(R.layout.activity_login);
41         //0. Registrar usuari de proves en local
42         PreferenceUtils.registrarUsuari(getApplicationContext(), "usuari", "123456"
43             );
44         //1. Settejem iconografia d'usuari/clau
45         ((TextView) findViewById(R.id.txvIconaUsuari)).setTypeface(FontManager.
46             getFA5SolidFont(getApplicationContext()));
47         ((TextView) findViewById(R.id.txvIconaClau)).setTypeface(FontManager.
48             getFA5SolidFont(getApplicationContext()));
49         //2. Recuperem EditText usuari i clau
50         this.editTextUsuari = (TextInputEditText) findViewById(R.id.edtUsuari);
51         this.editTextClau = (TextInputEditText) findViewById(R.id.edtClau);
52         ((Button) findViewById(R.id.btnIniciaSessio)).setOnClickListener(new View.
53             OnClickListener() {
54                 @Override
55                 public void onClick(View view) {
56                     LoginTask loginTask = new LoginTask(editTextUsuari.getText().
57                         toString(), editTextClau.getText().toString());
58                     loginTask.execute();
59                 }
60             });
61     }
62 }

```

La funció onCreate, bàsicament mostra la vista de la layout de login, carregant les icones d'usuari i de contrasenya, registra un usuari que serà el de proves pel nostre prototip, i el listener s'espera fins que algú polsa el botó inicia sessió.

Quan el listener rep que algú ha clicat el botó, amb segon pla comprova si les credencials són correctes, en cas contrari s'obrirà un missatge d'error de credencials i esperarà un altre cop que algú polsi el botó.

En cas d'haver-hi introduït l'usuari i la contrasenya vàlida, ens portarà a la següent activitat "scrollingActivity.class". Podem veure com està programat en la següent funció:

```

56
57 private class LoginTask extends AsyncTask<Void, Void, Boolean> {
58     private String usuari;
59     private String clau;
60     public LoginTask(String usuari, String clau) {
61         this.usuari = usuari;
62         this.clau = clau;
63     }
64     @Override
65     protected void onPreExecute() {
66         super.onPreExecute();
67     }
68     @Override
69     protected Boolean doInBackground(Void... voids) {
70         return LoginManager.validarUsuariClau(getApplicationContext(), this.
71             usuari, this.clau);
72     }
73     @Override
74     protected void onPostExecute(Boolean b) {
75         super.onPostExecute(b);
76     }
77 }

```

```

75         if (b) {
76             //Usuari loggejat correctament
77             Intent intent = new Intent(LoginActivity.this, ScrollingActivity.
78                 class);
79             startActivity(intent);
80             finish();
81         } else {
82             AlertDialog.Builder builder = new AlertDialog.Builder(LoginActivity
83                 .this);
84             builder.setTitle(getString(R.string.usuari_clau_incorrectes));
85             builder.setMessage(getString(R.string.
86                 usuari_clau_incorrectes_missatge));
87             builder.setPositiveButton(getString(R.string.acceptar), new
88                 DialogInterface.OnClickListener() {
89                 @Override
90                 public void onClick(DialogInterface dialogInterface, int i) {
91                 }
92             });
93             AlertDialog dialog = builder.create();
94             dialog.show();
95         }
96     }
97 }

```

### 9.2.3.2 scrollingActivity.class

Aquesta activitat és una mica més complexa que l'anterior, només d'entrar ens crea la layout amb tots els passadissos, estanteries del magatzem en el qual accedim. També ens crea i ens mostra el menú desplegable de l'aplicació.

```
1 private void setupMagatzem()
```

Aquesta funció simplement ens genera tota l'estructura de la vista, tant passadissos com estanteries, i posant-li les etiquetes dels seus noms al cim, també crea els listeners de cada botó de les estanteries, per si polsem al cim d'una s'activi l'activitat de registrar producte.

```
1 private void setupAppMenu()
```

SetupAppMenu ens genera tot el menú de l'aplicació, amb les diferents opcions, i quan s'obre el menú té un listener esperant que algú polsi una de les opcions cridant la funció corresponent per cada opció del menú.

Aquesta activitat ha estat reutilitzada per diferents activitats i pot ser "executada" de 3 formes diferents. Tenim una variable que ens controla en quin estat entra aquesta activitat, depenguen del mode en el que entra aquesta activitat actua diferent per cada cas. Els 3 casos són els següents:

1. Normal -> Simplement veiem el magatzem, com a pantalla principal amb el menú de totes les funcions que ens permet l'aplicació.
2. Selecció -> Quan entrem en el mapa venint de l'opció de registrar paquet, i necessitem escollir una ubicació pel nostre paquet.
3. Localització -> Quan venim de buscar un paquet, i s'ha seleccionat un per veure la seva localització en el mapa, on ens pinta l'estanteria de color verd.

---

Comprova en quin mode entra aquesta activitat amb la funció:

```
2 private void checkMode(Intent intent) {
3     Bundle bundle = intent.getExtras();
4     if (bundle != null) {
5         int launchModeValue = bundle.getInt(CommonUtils.LAUNCH_MODE_KEY, 0);
6         switch (launchModeValue) {
7             case CommonUtils.LAUNCH_MODE_NORMAL:
8                 LAUNCH_MODE = LaunchMode.NORMAL;
9                 break;
10            case CommonUtils.LAUNCH_MODE_SELECT:
11                LAUNCH_MODE = LaunchMode.SELECT;
12                break;
13            case CommonUtils.LAUNCH_MODE_LOCALIZE:
14                LAUNCH_MODE = LaunchMode.LOCALIZE;
15                ID_ESTANTERIA_LOCALIZE = bundle.getInt(CommonUtils.
16                    BUNDLE_KEY_ESTANTERIA_ID, 0);
17                ID_PRESTATGE_LOCALIZE = bundle.getInt(CommonUtils.
18                    BUNDLE_KEY_PRESTATGE_ID, 0);
19                break;
20            default:
21                LAUNCH_MODE = LaunchMode.UNSET;
22                break;
23        }
24    }
25 }
```

Ja que tenim el menú en aquesta activitat, també tenim definides les funcions que ens porten a les diferents activitats, com són la de registrar producte, eliminar producte, localitzar producte.

A l'entrar registrar producte pot ser que hi entrem perquè s'ha seleccionat una estanteria i prestatge, o pot ser que s'hi ha entri des del menú, per tant li passem per paràmetre la selecció d'estanteria ens cridarà l'activitat de seleccionar un prestatge, i si és NULL vol dir que s'ha entrat des del menú i ens llançarà l'activitat de registrar producte sense cap localització seleccionada, com es pot veure en el següent codi:

```
23 private void registrarProducte(@Nullable Estanteria estanteria) {
24     if (estanteria == null) {
25         Intent intent = new Intent(ScrollingActivity.this,
26             RegistrarProducteActivity.class);
27         Bundle bundle = new Bundle();
28         bundle.putInt(CommonUtils.BUNDLE_KEY_ESTANTERIA_ID, 0);
29         bundle.putInt(CommonUtils.BUNDLE_KEY_PRESTATGE_ID, 0);
30         intent.putExtras(bundle);
31         startActivity(intent);
32     } else {
33         Intent intent = new Intent(ScrollingActivity.this, PrestatgesActivity.
34             class);
35         Bundle bundle = new Bundle();
36         bundle.putInt(CommonUtils.BUNDLE_KEY_ESTANTERIA_ID, (estanteria == null
37             ? 0 : estanteria.getId()));
38         bundle.putInt(CommonUtils.LAUNCH_MODE_KEY, CommonUtils.
39             LAUNCH_MODE_NORMAL);
40         intent.putExtras(bundle);
41         startActivity(intent);
42     }
43 }
```

---

### 9.2.3.3 RegistrarProducteActivity.java

Com a totes les activitats tenim la funció `onCreate()` que ens genera la vista de la layout per pantalla.

```
40 private void seleccionarUbicacio() {
41     Intent intent = new Intent(RegistrarProducteActivity.this,
42         ScrollingActivity.class);
43     Bundle bundle = new Bundle();
44     bundle.putInt(CommonUtils.LAUNCH_MODE_KEY, CommonUtils.LAUNCH_MODE_SELECT);
45     intent.putExtras(bundle);
46     startActivityForResult(intent, CommonUtils.MAPA_MAGATZEM_REQUEST_CODE);
47 }
```

En la funció anterior es pot comprovar com tornem a anar a l'activitat anterior (`ScrollingActivity`), però amb un altre mode, el de selecció, ja que s'havia entrat a registrar producte des del menú i no teníem cap localització pel paquet seleccionada. Això ens permetrà entrar en el mapa i seleccionar una ubicació per aquest paquet.

Si anem a llegir el codi NFC del paquet ens portarà a la següent activitat, per tal que el dispositiu mòbil detecti el xip.

En tenir totes les dades del paquet ens permetrà guardar-lo a la base de dades, aquesta funció és la que insereix el paquet a la base de dades amb totes les dades introduïdes. En cas que alguna està a null llança un error.

```
47 private void desarProducte() {
48     if (estanteria == null) {
49         //Mostrem error a la ubicacio
50         edtUbicacio.setError(getString(R.string.es_necessari_establir_ubicacio)
51             );
52         return;
53     }
54     if (TextUtils.isEmpty(edtCodiNFC.getText())) {
55         //Mostrar error codi NFC
56         edtCodiNFC.setError(getString(R.string.el_codi_nfc_es_obligatori));
57         return;
58     }
59     if (TextUtils.isEmpty(edtNomProducte.getText())) {
60         //Mostrar error nom
61         edtNomProducte.setError(getString(R.string.el_nom_es_obligatori));
62         return;
63     }
64     Producte producte = new Producte(
65         0,
66         edtCodiNFC.getText().toString(),
67         edtNomProducte.getText().toString(),
68         edtAdrecaProducte.getText().toString(),
69         prestatge.getId()
70     );
71     MagatzemDb magatzemDb = MagatzemDb.getInstance(this);
72     magatzemDb.producteDao().insert(producte);
73     this.onBackPressed();
74 }
```

---

### 9.2.3.4 NfcReaderActivity.java

Aquesta activitat és a la que li dono més importància, ja que al final el projecte, tracta dels xips NFC, en aquest apartat quedarà explicada en detall.

Primer de tot s'ha de configurar en el fitxer "AndroidManifest.xml" perquè l'aplicació pugui detectar els xips NFC correctament.

En el següent tall de codi extret d'aquest fitxer, gràcies això l'aplicació li permet llegir els xips NFC, i el "mimeType" és la configuració de la informació que han de tenir els xips NFC per tal de ser llegits.

```
75 <intent-filter>
76     <action android:name="android.nfc.action.NDEF_DISCOVERED" />
77
78     <data android:mimeType="@string/mimetypenfc" />
79 </intent-filter>
80 <meta-data
81     android:name="android.nfc.action.TECH_DISCOVERED"
82     android:resource="@xml/nfc_tech_filter" />
```

L'etiqueta està configurant un fitxer extern anomenat "nfc\_tech\_filter" que ens configura els xips que poden ser llegits.

En el fitxer xml actualment només hi ha configurat el tipus de xips que utilitzem per fer la demo, ja que així no carreguem el codi. Però tenim una llista de molts xips compatibles amb els sensors dels dispositius mòbils actuals que estan adaptats per ser llegits.

```
83 <?xml version="1.0" encoding="utf-8"?>
84 <resources xmlns:xliff="urn:oasis:names:tc:xliff:document:1.2">
85 <tech-list>
86     <tech>android.nfc.tech.Ndef</tech>
87 </tech-list>
88 </resources>
89
90 <!--
91 La llista de les diferents tecnologies NFC acceptades:
92
93 <tech-list>
94     <tech>android.nfc.tech.IsoDep</tech>
95     <tech>android.nfc.tech.NfcA</tech>
96     <tech>android.nfc.tech.NfcB</tech>
97     <tech>android.nfc.tech.NfcF</tech>
98     <tech>android.nfc.tech.NfcV</tech>
99     <tech>android.nfc.tech.Ndef</tech>
100    <tech>android.nfc.tech.NdefFormatable</tech>
101    <tech>android.nfc.tech.MifareClassic</tech>
102    <tech>android.nfc.tech.MifareUltralight</tech>
103 </tech-list>
104 -->
```

Un cop configurat tot això ja està llest per implementar l'activitat de llegir NFCs. En la funció "onCreate()" mostrem la layout com a totes les activitats explicades anteriorment, un cop llançada la layout es comprova que el dispositiu tingui sensor NFC, en cas de no tenir-ne ens sortirà un error de què el mòbil en qüestió no és apta per la nostra aplicació.

---

Si el dispositiu és apta comprova que el sensor NFC estigui activat pel correcte funcionament de l'aplicació, si no estigués activat ens demanaria que l'activéssim. Codi del "onCreate()":

```
105 @Override
106     protected void onCreate(Bundle savedInstanceState) {
107         super.onCreate(savedInstanceState);
108         setContentView(R.layout.activity_nfc_reader);
109         //1. Amaguem l'ActionBar
110         getSupportActionBar().hide();
111         //2. Comprovem si tenim un dispositiu compatible amb NFC
112         nfcAdapter = NfcAdapter.getDefaultAdapter(this);
113         if (nfcAdapter == null) {
114             AlertDialog.Builder builder = new AlertDialog.Builder(NfcReaderActivity
115                 .this);
116             builder.setCancelable(false);
117             builder.setTitle(getString(R.string.dispositiu_sense_NFC));
118             builder.setMessage(getString(R.string.aquest_dispositiu_no_NFC));
119             builder.setPositiveButton(getString(R.string.acceptar), new
120                 DialogInterface.OnClickListener() {
121                 @Override
122                 public void onClick(DialogInterface dialogInterface, int i) {
123                     NfcReaderActivity.this.finish();
124                 }
125             });
126             AlertDialog dialog = builder.create();
127             if (dialog != null) dialog.show();
128             return;
129         } else if (!nfcAdapter.isEnabled()) {
130             AlertDialog.Builder builder = new AlertDialog.Builder(NfcReaderActivity
131                 .this);
132             builder.setCancelable(false);
133             builder.setTitle(getString(R.string.NFC_deshabilitat));
134             builder.setMessage(getString(R.string.has_dhabilitar_NFC));
135             builder.setPositiveButton(getString(R.string.configuracio), new
136                 DialogInterface.OnClickListener() {
137                 @Override
138                 public void onClick(DialogInterface dialogInterface, int i) {
139                     startActivity(new Intent(android.provider.Settings.
140                         ACTION_NFC_SETTINGS));
141                     NfcReaderActivity.this.finish();
142                 }
143             });
144             AlertDialog dialog = builder.create();
145             if (dialog != null) dialog.show();
146             return;
147         }
148         //3. Recuperem el layout pare
149         parentLayout = (LinearLayout) findViewById(R.id.parentLayout);
150         //4. Tractem l'intent, si cal
151         handleIntent(getIntent());
152     }
```

Quan llegeix un xip NFC comprova que la tecnologia del xip i el tipus d'informació que llegeix siguin el configurat prèviament, en cas contrari surt un error avisant que estem llegint un xip erroni. Si el xip és de la tecnologia configurada prèviament, extreu l'objecte tipus "tag" i s'inicia una tasca asíncrona per tractar aquest "tag". Aquesta funcionalitat que comprova la tecnologia i la informació del xip NFC està

---

implementada en la funció "handleIntent()":

```
148 private void handleIntent(Intent intent) {
149     String action = intent.getAction();
150     if (NfcAdapter.ACTION_NDEF_DISCOVERED.equals(action)) {
151         String type = intent.getType();
152         if (MIME_TEXT_PLAIN.equals(type)) {
153             Tag tag = intent.getParcelableExtra(NfcAdapter.EXTRA_TAG);
154             (new NdefReaderTask()).execute(tag);
155         } else {
156             Log.d(TAG, "Wrong mime type: " + type);
157         }
158     } else if (NfcAdapter.ACTION_TECH_DISCOVERED.equals(action)) {
159         // In case we would still use the Tech Discovered Intent
160         Tag tag = intent.getParcelableExtra(NfcAdapter.EXTRA_TAG);
161         String[] techList = tag.getTechList();
162         String searchedTech = Ndef.class.getName();
163         for (String tech : techList) {
164             if (searchedTech.equals(tech)) {
165                 (new NdefReaderTask()).execute(tag);
166                 break;
167             }
168         }
169     }
170 }
```

En la mateixa classe.java tenim unes funcions "onResume" i "onPause" que ens permeten aturar i tornar a arrencar l'activitat de lectura de NFC, per no tenir els listeners sempre oberts quan estàs a altres activitats, i "aturar" el sensor i només obrir-lo quan es llança una activitat de lectura dels xips.

Tenim una classe anomenada "NdefReaderTask" que és l'encarregada de tractar els objectes "tag" descrits anteriorment, la primera funció més important d'aquesta classe és la "doInBackground()", és una mica complicada d'explicar el seu funcionament, bàsicament tracta l'objecte "tag" i n'extreu la informació cap a un objecte NdefRecord, que és un vector amb la informació tractada.

Un cop llegida i guardada en l'objecte NdefRecord s'envia cap a la funció següent "readText()", converteix aquest objecte amb un String, de manera que és llegible per nosaltres i l'aplicació el pot tractar més fàcilment.

```
171 protected String doInBackground(Tag... params) {
172     Tag tag = params[0];
173     Ndef ndef = Ndef.get(tag);
174     if (ndef == null) {
175         // NDEF is not supported by this Tag.
176         return null;
177     }
178     NdefMessage ndefMessage = ndef.getCachedNdefMessage();
179     NdefRecord[] records = ndefMessage.getRecords();
180     for (NdefRecord ndefRecord : records) {
181         if (ndefRecord.getTnf() == NdefRecord.TNF_WELL_KNOWN && Arrays.
182             equals(ndefRecord.getType(), NdefRecord.RTD_TEXT)) {
183             try {
184                 return readText(ndefRecord);
185             } catch (UnsupportedEncodingException e) {
186                 Log.e(TAG, "Unsupported Encoding", e);
187             }
188         }
189     }
190 }
```



```

187         }
188     }
189     return null;
190 }
191
192 private String readText(NdefRecord record) throws
193     UnsupportedEncodingException {
194     byte[] payload = record.getPayload();
195     // Get the Text Encoding
196     String textEncoding = ((payload[0] & 128) == 0) ? "UTF-8" : "UTF-16";
197     // Get the Language Code
198     int languageCodeLength = payload[0] & 0063;
199     // Get the Text
200     return new String(payload, languageCodeLength + 1, payload.length -
        languageCodeLength - 1, textEncoding);
}

```

Per acabar de detallar tot el procés que fa l'activitat del "NfcReaderActivity", l'últim mètode segons si s'ha pogut llegir bé el xip NFC envia les dades a través del String format, cap a l'activitat d'on ens ha portat a fer la lectura. Si no s'han pogut llegir les dades, simplement donem un missatge d'error perquè l'usuari ho vegi.

```

202     @Override
203     protected void onPostExecute(String result) {
204         if (result != null) {
205             Intent intent = new Intent();
206             Bundle bundle = new Bundle();
207             bundle.putString(CommonUtils.BUNDLE_KEY_CODI_NFC, result);
208             intent.putExtras(bundle);
209             NfcReaderActivity.this.setResult(Activity.RESULT_OK, intent);
210             NfcReaderActivity.this.finish();
211         } else {
212             Snackbar.make(parentLayout, getString(R.string.
213                 codi_nfc_no_interpretat), Snackbar.LENGTH_LONG)
214                 .setAction(getString(R.string.acceptar), null)
215                 .show();
216         }
}

```

### 9.2.3.5 ProductesActivity.java

Aquesta activitat també té diversos modes per entrar a ella, la primera és la de buscar un paquet, l'altre és la d'eliminar un paquet, s'utilitza la mateixa activitat.

A l'accedir a l'activitat ens genera la layout de la pantalla segons en quin dels 2 modes entrem. L'activitat té un listener que a mesura que s'escriu el nom, va eliminant de la llista els paquets que no contenen la cadena de caràcters. També té un listener per si es vol llegir el codi NFC del paquet i ens surti directe el paquet que volem.

Si estem en mode de buscar paquet i en seleccionem un de la llista ens crida la següent funció, que ens portarà a l'Activitat "ScrollingActivity" amb mode localització.

217

```

218     private void mostrarUbicacioProducte(int prestatgeId) {
219         Prestatge prestatge = MagatzemDb.getInstance(ProductesActivity.this).
                prestatgeDao().getByPk(prestatgeId);
220         Intent intent = new Intent(ProductesActivity.this, ScrollingActivity.class)
                ;
221         Bundle bundle = new Bundle();
222         bundle.putInt(CommonUtils.LAUNCH_MODE_KEY, CommonUtils.LAUNCH_MODE_LOCALIZE
                );
223         bundle.putInt(CommonUtils.BUNDLE_KEY_PRESTATGE_ID, (prestatge == null ? 0 :
                prestatge.getId()));
224         bundle.putInt(CommonUtils.BUNDLE_KEY_ESTANTERIA_ID, (prestatge == null ? 0
                : prestatge.getEstanteriaId()));
225         intent.putExtras(bundle);
226         startActivity(intent);
227     }

```

En el mode d'esborrar un producte quan seleccionem un o varis paquets de la llista ens cridarà la funció:

```

229     private void eliminarProductesFiltrats() {
230         MagatzemDb magatzemDb = MagatzemDb.getInstance(this);
231         for (Producte producte : productes)
232             if (producte.getFiltrat()) magatzemDb.producteDao().delete(producte);
233         productes = magatzemDb.producteDao().getAll();
234         edtCodiNFC.setText("");
235     }
236 }

```

Com veiem ens recorrem la llista de productes que tenim seleccionats, fent un delete a la base de dades, per tal de borrar-los del nostre prestatge perquè ja passen a ser repartit al client.

### 9.2.3.6 PrestatgesActivity.java

La layout d'aquesta activitat és l'única que utilitza un recycleview, ja que guardar una layout de N prestatges amb la informació dels paquets guardats per cada estanteria seria molt costós i ocuparia memòria que ens podem estalviar perquè l'APP sigui més lleugera. Quan es crida aquesta activitat el recycleview crea els N prestatges mostrant la layout.

Igual que l'activitat del mapa del magatzem també 3 modes diferents, els mateixos perquè els prestatges també formen part el magatzem i actuen igual.

En seleccionar una estanteria ens cridarà l'activitat de selecció de prestatges, si entrem en mode "SELECT" vol dir que estàvem registrant un producte sense localització, i ara l'estem seleccionant posteriorment.

Si entrem en mode normal, vol dir que ens movíem pel nostre mapa del magatzem, i s'ha seleccionat una localització on es vol registrar un producte.

```

237     @Override
238     public void prestatgeWasSelected(Prestatge prestatge) {
239         Intent intent;
240         Bundle bundle;
241         switch (LAUNCH_MODE) {
242             case SELECT:
243                 intent = new Intent();

```

---

```

244         bundle = new Bundle();
245         bundle.putInt(CommonUtils.BUNDLE_KEY_ESTANTERIA_ID, ID_ESTANTERIA);
246         bundle.putInt(CommonUtils.BUNDLE_KEY_PRESTATGE_ID, (prestatge ==
                null ? 0 : prestatge.getId()));
247         intent.putExtras(bundle);
248         PrestatgesActivity.this.setResult(Activity.RESULT_OK, intent);
249         PrestatgesActivity.this.finish();
250         break;
251     case NORMAL:
252         intent = new Intent(PrestatgesActivity.this,
                RegistrarProducteActivity.class);
253         bundle = new Bundle();
254         bundle.putInt(CommonUtils.BUNDLE_KEY_ESTANTERIA_ID, ID_ESTANTERIA);
255         bundle.putInt(CommonUtils.BUNDLE_KEY_PRESTATGE_ID, (prestatge ==
                null ? 0 : prestatge.getId()));
256         intent.putExtras(bundle);
257         startActivity(intent);
258         PrestatgesActivity.this.finish();
259         break;
260     default:
261         break;
262     }
263 }

```

---

## 10 Implantació i resultats

### 10.1 Resultats: Demostració dels requeriments

En aquest apartat demostrarem que tots els requeriments inicials que havia de tenir l'aplicació, funcionen correctament. És una mica complicat la demostració de què cada un funciona, per tant la millor forma que he trobat de demostrar-ho sense fer un demo complet, és amb fotos de l'aplicació fent cada requeriment.

#### 10.1.1 Instal·lació

Actualment l'aplicació prototip no es pot instal·lar sense tenir el codi font del projecte, perquè actualment l'app es troba en desenvolupament, no la podem trobar al Google Store

Per fer la instal·lació de l'aplicació s'ha de tenir el SO Android en mode desenvolupador, en cada dispositiu pot ser diferent a l'hora de configurar el mode de desenvolupador. En l'exemple d'instal·lació s'ha agafat un model Huawei P20.

Per configurar el mode de desenvolupador, s'ha d'entrar a "*Ajustes-> Acerca del teléfono*" i anar polsant diversos cops a "*Numero de compilación*", ens demanarà que posem el codi de seguretat o el patró de desbloqueig, un cop fet això tenim el dispositiu configurat per poder instal·lar l'aplicació. Com podem observar en la següent foto ens surt que ja estem en mode de desenvolupador, el nostre dispositiu ja està preparat per instal·lar-hi el WareHouse-sync.



Figura 18: Mode de desenvolupador activat.

Connecte'm via cable USB el dispositiu en el PC, de manera que l'Android Studio el detecti, canviem l'opció de telèfon virtual i seleccionem el nostre per executar el programa, donem a compilar o el botó de play de color verd. Ens podem guiar pel menú en la següent imatge, ens mostra tots els dispositius que ens detecta l'Android Studio, seleccionem en aquest exemple el Huawei.

Després d'uns processos de compilació de l'Android Studio, ens hauria de sortir l'aplicació ja instal·lada en el nostre dispositiu Huawei.

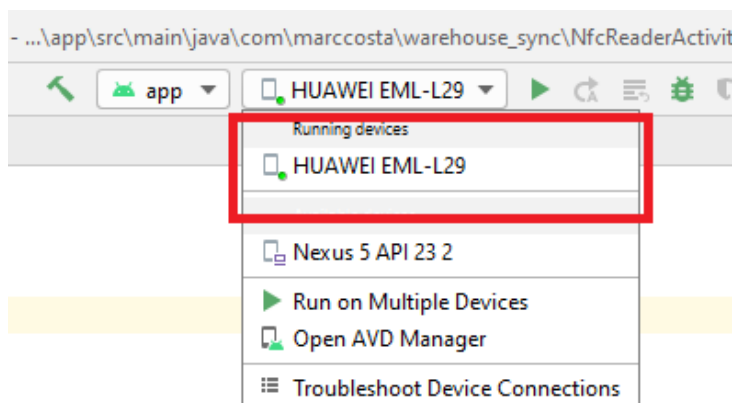
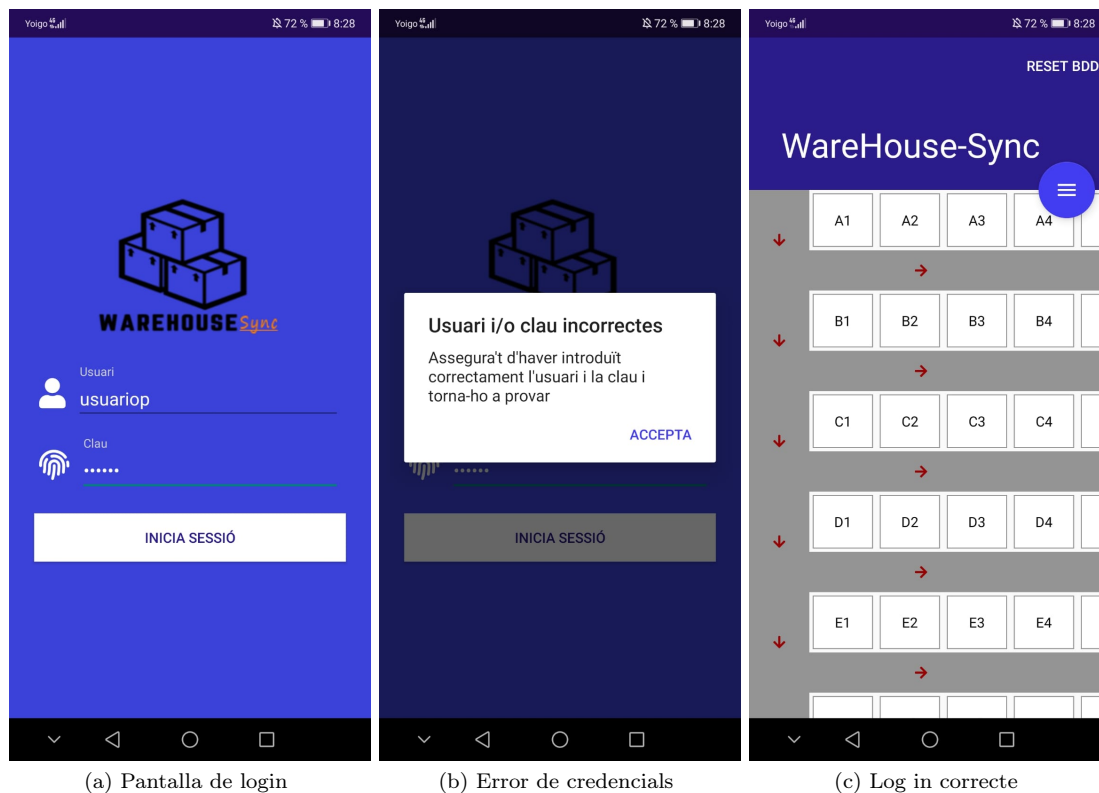


Figura 19: Selecció del dispositiu a l'Android Studio

### 10.1.1.1 Log-in

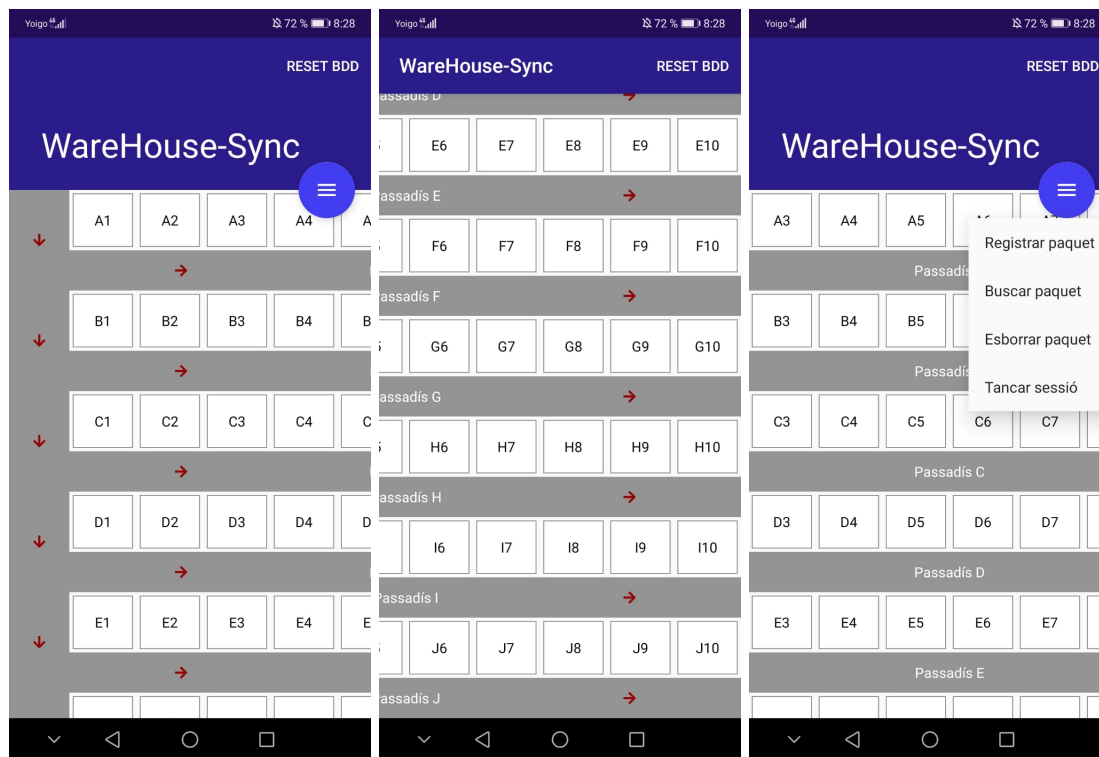
El primer requeriment que hauria de funcionar és el de què un usuari pugui accedir dins de l'aplicació amb les seves credencials, provem tots els casos:



- L'usuari introdueix les credencials per poder accedir a l'aplicació.
- Com podem veure, l'usuari s'ha equivocat en les credencials i ens surt un missatge d'error, fer-nos repetir el procés d'identificació.
- Finalment, un cop les credencials correctes, entrem a l'aplicació veiem ja el magatzem, i poden utilitzar l'aplicació amb total normalitat.

### 10.1.1.2 Visualitzar magatzem, fer scroll i visualitzar el menú

Un cop loguejat amb les credencials correctes, visualitzem la pantalla principal de l'aplicació, anem a veure els diferents requeriments d'aquesta pantalla:



(d) Visualització del mapa del magatzem

(e) Scrolling pel mapa

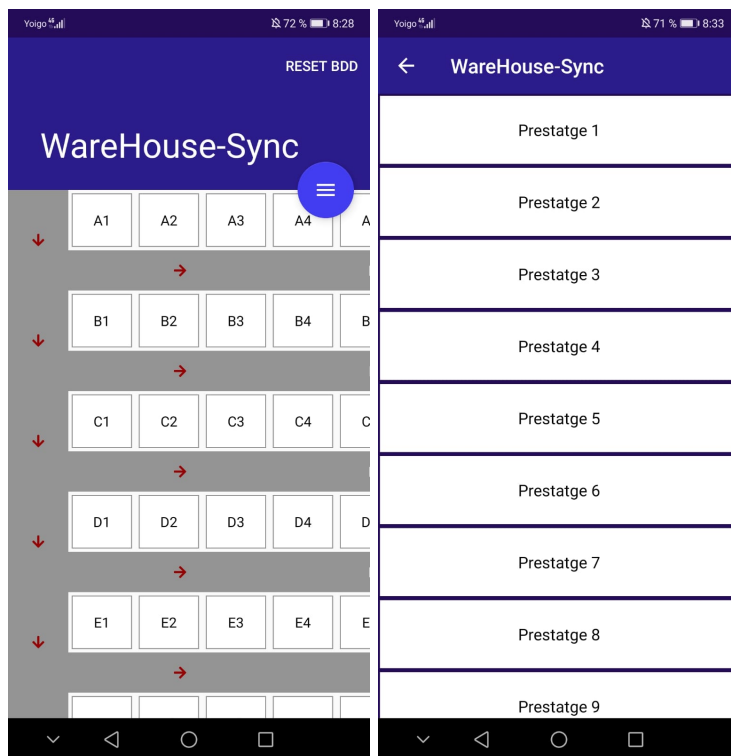
(f) Menú de l'aplicació

- d) L'usuari visualitza tots els passadissos i les estanteries de cada un d'ells.
- e) Podem fer scroll pel mapa visualitzant la resta del magatzem que no es pot veure en pantalla, permeten accedir a totes les parts del magatzem.
- f) Tenim el botó de menú, que un cop polsat se'ns desplega, i podem accedir a les diferents funcions que permet el Warehouse-sync.

---

### 10.1.1.3 Prèmer una estanteria per veure els prestatges

Anem a comprovar que es pugui accedir a dins de les estanteries per visualitzar els seus prestatges.

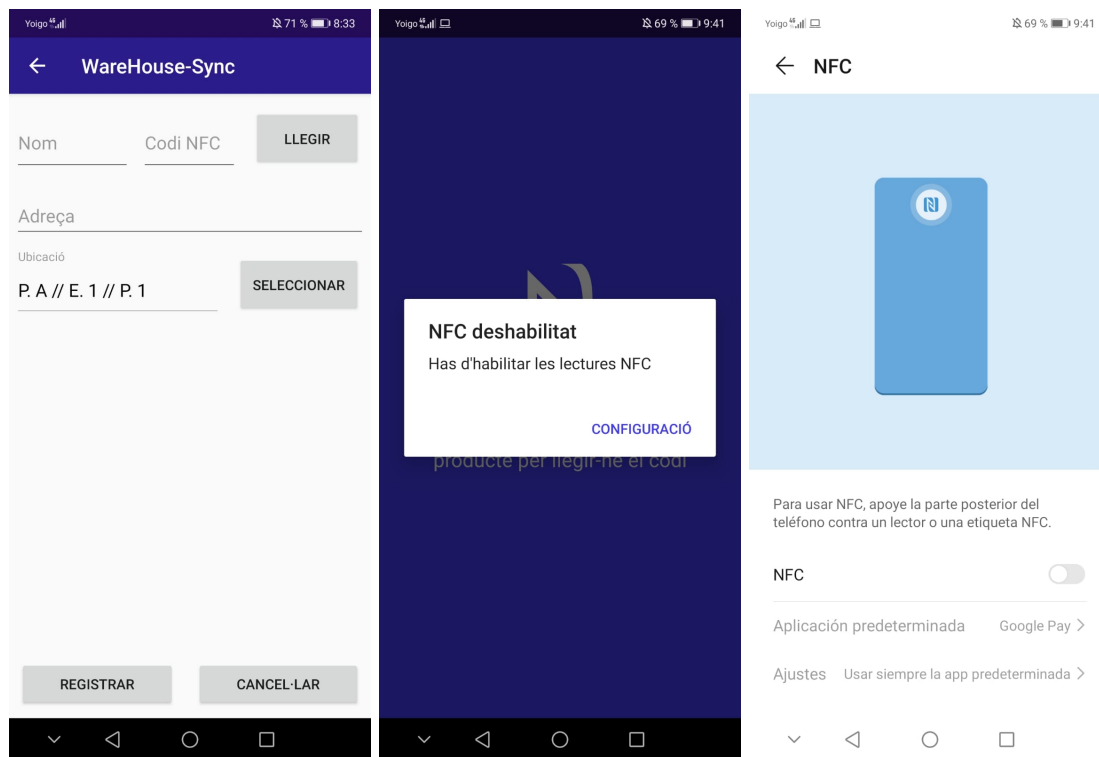


(g) Visualització del mapa del magatzem (h) Prestatges de l'estanteria seleccionada

- g) L'usuari visualitza tots els passadissos i les estanteries del magatzem.
- h) Si se selecciona una estanteria, sortiran els seus prestatges, en cas de seleccionar un prestatge ens portarà al requeriment de registrar un paquet amb localització prèviament seleccionada.

#### 10.1.1.4 Llegir xip NFC

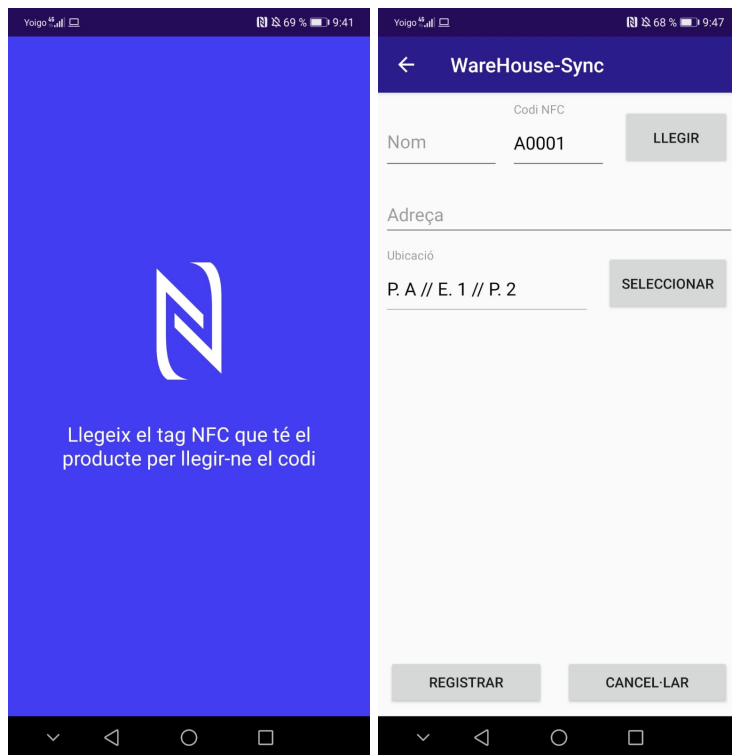
Podem llegir NFC a diferents pantalles de la nostra aplicació, en registrar paquet o bé a eliminar paquet. Anem a comprovar si l'aplicació ens llegeix els xips NFC.



(i) Finestra registrar paquet per fer la lectura NFC (j) Avís que no tenim NFC activat (k) Ens porta activar el lector NFC

- i) Ens disposem a fer la lectura d'un xip NFC a partir de la pantalla de registrar un paquet.
- j) Quan intentem llegir ens avisa que no tenim el lector NFC del mòbil activat.
- k) Si polsem configuració en la pantalla de l'avís, ens porta a configuració del nostre dispositiu per tal de connectar el sensor NFC.





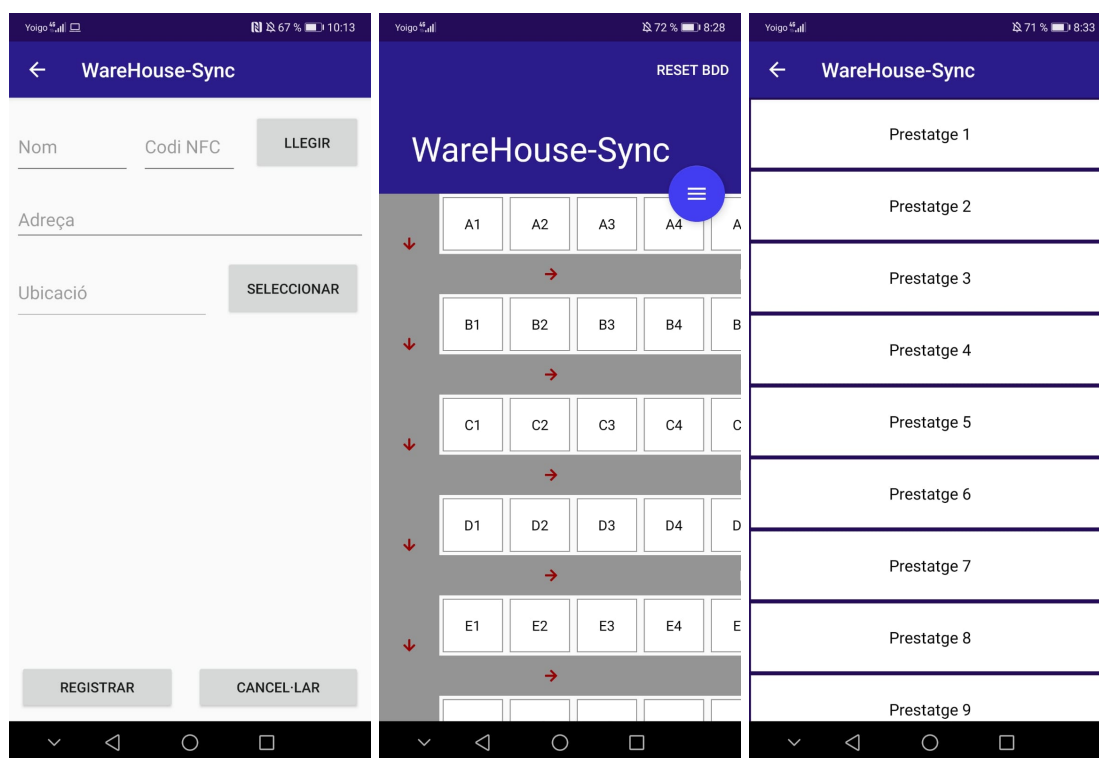
(l) Espera detectar un xip NFC (m) Informació del xip NFC detectat

- l) Un cop activat el lector NFC, l'aplicació continuarà el procés de lectura del xip.
- m) Finalment quan detecta el xip NFC, processa la informació del tag i l'insereix a la pantalla de registre en forma de string.

### 10.1.1.5 Registrar un paquet i buscar paquet

En aquest apartat comprovarem el correcte funcionament de la pantalla per registrar un paquet.

**Important:** Cal recordar que es pot accedir aquesta funcionalitat des del menú, fet que ens farà no tenir cap localització en el magatzem pel paquet que estem registrant, i s'haurà de seleccionar a posteriori. O l'altra manera és havent seleccionat estanteria i prestatge del magatzem, ja tindrem la localització seleccionada.

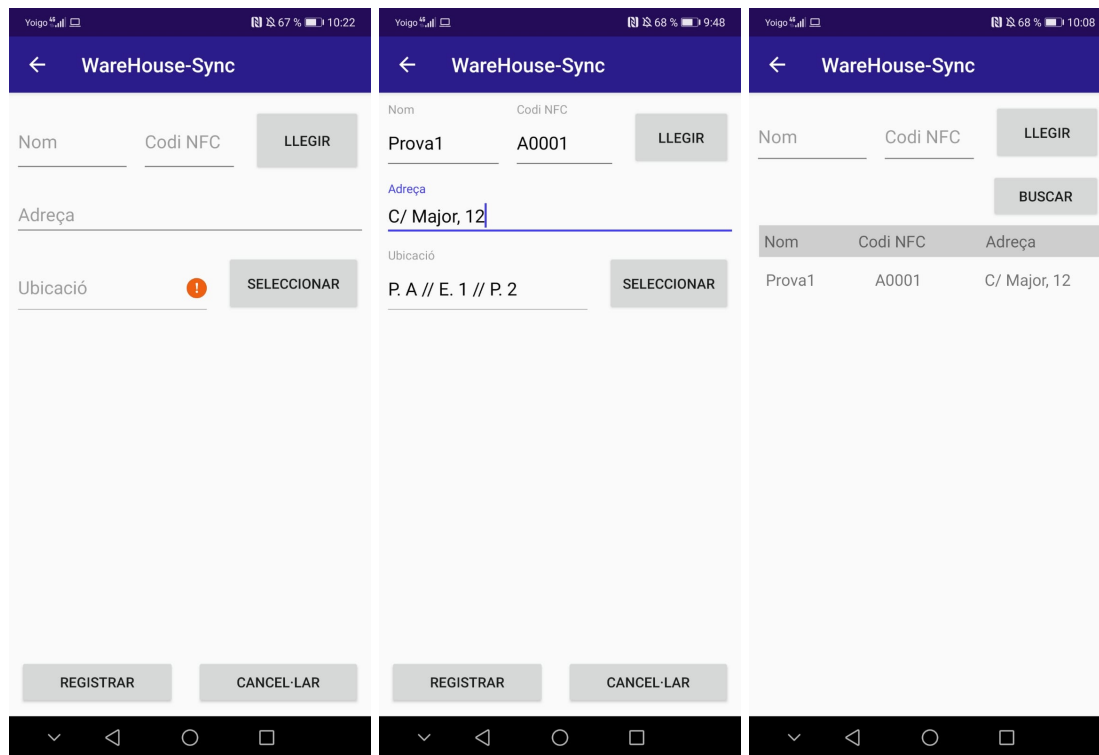


(n) Pantalla de registrar entrat des del menú

(o) Selecció de localització

(p) Selecció de prestatge

- n) Entrem des del menú a la pantalla de registrar paquet, no tenim cap localització seleccionada.
- o) Polsem seleccionar i ens deixa escollir dins del magatzem l'estanteria.
- p) Polsem una estanteria i ens deixa escollir el prestatge en què guardarem el paquet.



(q) Error falten dades

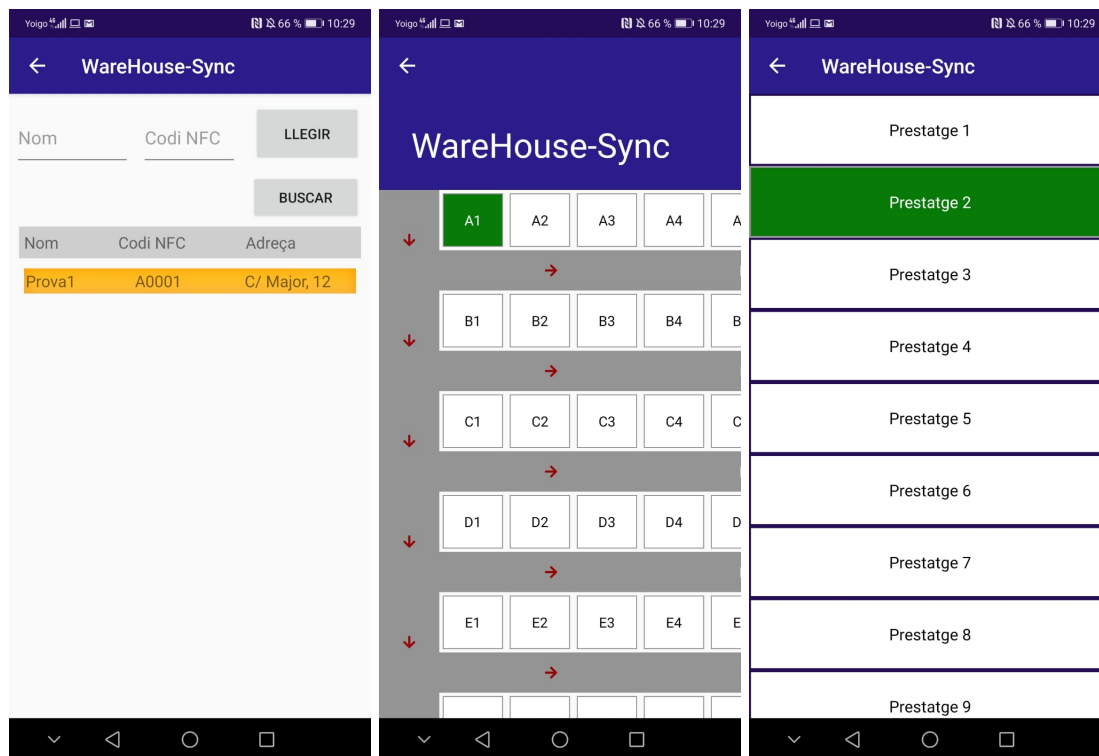
(r) Totes les dades entrades

(s) Pantalla buscar paquet

- q) Si no entrem les dades del paquet i intentem registrar-lo, ens sortirà una icona indicant les dades que falten per introduir.
- r) En tenir totes les dades introduïdes, podem procedir a registrar-lo.
- s) Com podem observar a la pantalla de buscar paquet, ens funciona correctament mostrant tots els paquets registrats, actualment només 1. En cas que el seleccionéssim ens portaria al següent requeriment, localitzar-lo en el mapa del magatzem.

### 10.1.1.6 Localitzar paquet en el magatzem

Aquest requeriment ens hauria de marcar la localització del paquet dins del mapa del magatzem.



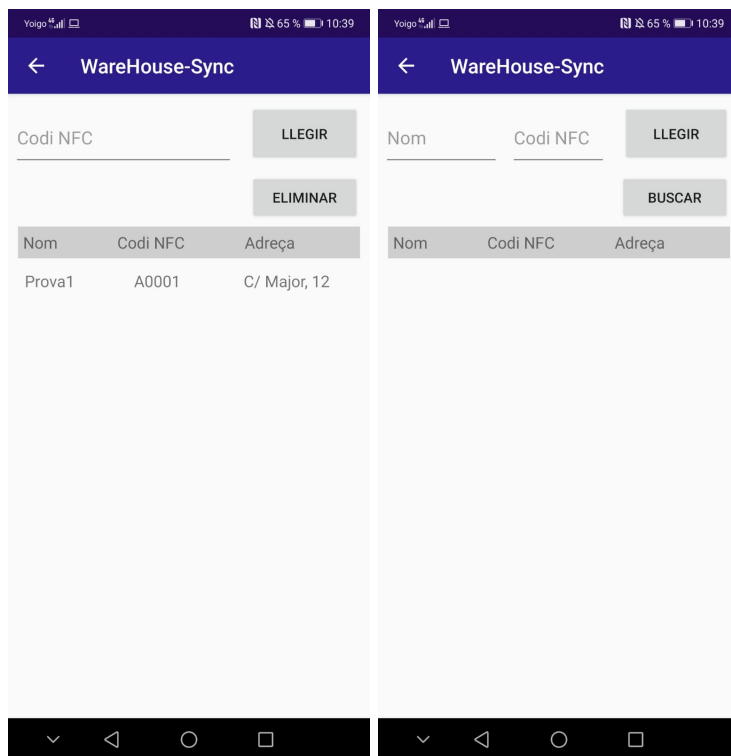
(t) Seleccionem un paquet a buscar (u) Localització en el mapa del ma- (v) Localització del prestatge dins gatzem on es troba l'estanteria

- t) Seleccionem un paquet en la pantalla de buscar paquet, per tal de saber la seva localització dins del magatzem.
- u) Un cop seleccionat el paquet que volem saber la seva localització, podem veure a quin passadís i quina estanteria el podem trobar, es marca pintat de color verd.
- v) Seleccionem l'estanteria, i ens mostra a quin prestatge el podem trobar.

---

### 10.1.1.7 Esborrar un paquet del magatzem

El requeriment era esborrar un paquet de la seva localització i de la base de dades.



(w) Pantalla d'esborrar paquet

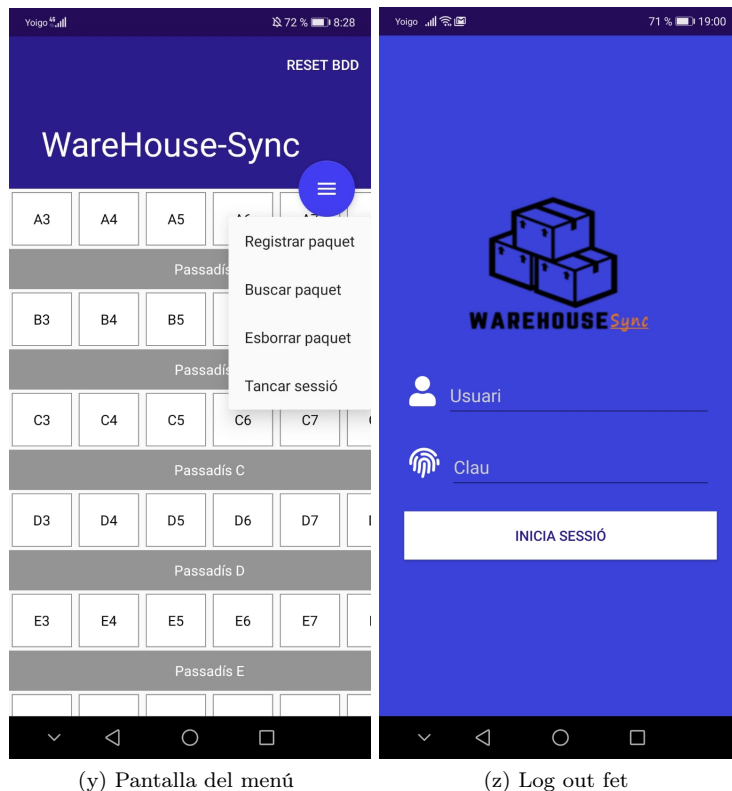
(x) Paquet esborrat de la base de dades

- w) La pantalla d'esborrar un paquet és molt semblant a la de buscar un paquet, però s'ha d'anar amb molt de compte de no esborrar cap per error, ja que no es pot tornar enrere.
- x) Quan el esborrem, ja no figura a la base de dades, per tant el paquet ha sortit del magatzem i està en estat d'entrega al client.

---

### 10.1.1.8 Sortir de l'aplicació

Anem a fer un log out per comprovar que ens surt de la connexió a l'aplicació, portar-nos a la pantalla de log in.



- y) Entrem el menú, seleccionem l'opció de log out.
- z) Es tanca la nostra sessió com a usuari de l'aplicació.

---

## 11 Conclusions

Com s'ha pogut veure al llarg de tot el treball de fi de grau s'ha complert l'objectiu principal que s'havia plantejat des d'un principi, al final s'ha aconseguit treure un producte per millorar la indústria de la logística, utilitzant una versió actualitzada del codi de barres i molt més eficient pels diferents avantatges respecte a la tecnologia utilitzada fins ara.

En ser un projecte més llarg que una simple pràctica de laboratori m'agradaria valorar l'experiència obtinguda en el transcurs del projecte, he aconseguit autodedicació, conceptes de planificació del temps i anàlisis dels pilars fonamentals per convertir una idea a un fet.

A part d'aconseguir l'objectiu final proposat en el plantejament del treball, a mesura que s'anava indagant en el tema sortien altres objectius més petits, que s'anaven proposant en el camí per tal d'assolir el principal objectiu. Vaig tenir una mica de desmotivació quan la primera idea implementada, que feia anys que volia implementar algun dispositiu així, no ha complert els objectius esperats, ja que la tecnologia NFC no era suficient per compatir amb els codis de barres a l'hora de "passar per caixa" en un supermercat, però possiblement d'aquí poc temps sigui viable aconseguir aquest propòsit, gràcies a la sortida d'una nova tecnologia que és molt semblant la tecnologia NFC millorada, l'Ultra-Wideband (UWB). Promet que es podrà regular la distància de lectura de 30cm-200m i serà molt fàcil de configurar, també que podrà llegir diferents tipus de xips.[?]

Al fet de no aconseguir els objectius secundaris a l'hora de plantejar la primera idea de desenvolupament, s'ha hagut de replantejar l'enfocament del treball buscant una altra manera de portar l'objectiu final a l'èxit. Ja tenia constància que podia passar, perquè el meu treball no anava amb una idea molt clara, sinó que estava més enfocat en la investigació del sector a veure com es podia millorar l'eficiència del treball en la logística i agilitzar els diferents processos que hi ha en els magatzems actuals, incorporant una nova tecnologia poc utilitzada com és la NFC.

També per culpa de la pandèmia no he aconseguit investigar com m'hagués agradat, la meua intenció era tornar a les empreses visitades anteriorment intentar vendre el producte final dissenyat, per veure les reaccions i les possibles millores que els clients reals voldrien, però estic content amb els resultats obtinguts.

Tot aquest projecte s'ha dut a terme amb molt d'esforç, ja que no havia programat mai amb Android, i era un llenguatge que em feia molt de respecte, però a l'entendre el funcionament de les activitats, i les vistes és un llenguatge molt organitzat i entengible fàcilment. M'ha servit per aprendre coses noves del llenguatge pel fet que he hagut d'obtenir diversos coneixements de programació d'Android, amb un autoaprenentatge a partir de fòrum, tutorials o webs especialitzades. Ha estat difícil aquest autoaprenentatge com que la carrera sempre ens han donat els apunts i les coses necessàries per travessar diferents obstacles proposats pels professors, en ser un projecte individual i molt especialitzat m'he hagut d'adaptar, aprendre el coneixement necessària per tirar endavant el projecte saltant els diferents problemes que m'anava trobant a l'hora de programar i dissenyar l'aplicació, i finalment assolir l'objectiu.

Cal insistir en que l'aplicació WareHouse-sync és un prototip i li queda moltes funcionalitats que se li podrien afegir, i arribar a ser una pionera dins del mercat. Gràcies al dedicar molt de temps en l'aplicació, també analitzar el mercat, he adquirit noves idees que podrien valorar més aquesta aplicació.

Valorar també tota l'experiència adquirida tant personal com professional, donar les gràcies al Tutor Antonio Bueno per la seva dedicació i paciència, a la universitat per donar-nos la llibertat d'enfocar el nostre TFG a qualsevol tema relacionat amb la carrera, fet que ens ajuda molt de cara a ser futurs Enginyers i dur a terme projectes que comencen essent una simple idea.

---

## 12 Treball futur

La meua idea és seguir amb el projecte desenvolupat per portar-la cap endavant, dissenyant un producte final.

Algunes idees de treballs futurs podrien ser:

1. **Tecnologia Ultra-WideBand** - Un possible treball seria investigar els costos i la seva eficiència respecte els NFC, per recuperar el meu primer prototip operatiu, i amb tots els objectius complerts.
2. **Millora del WareHouse-sync** - Es podria millorar la meua aplicació, fent que funcionés via internet, i gestionant l'entrega de paquets un cop el transportista l'acaba d'entregar al client. Millorant l'accessibilitat i l'usabilitat de l'app.
3. **Automatitzar un magatzem** - Un projecte més sofisticat i possiblement molt més car de presupost, però ja que estan sortint molts magatzems automatitzats, investigar com es podrien fer més eficients, i/o utilitzant els xips NFC per tal que el robot es localitzi dins del magatzem llegint 1 xip, etc...



---

## 13 Annex

### 13.1 Manual d'usuari

#### 13.1.1 Pantalla de login

A l'obrir la nostra aplicació "Warehouse-sync" ens sortirà una pantalla de login, com que és un producte destinat a empreses, no et pots registrar. En el prototip actual només tenim un magatzem treballant amb l'aplicació, tenen com usuari=usuari i com a contrasenya="123456", un cop entrem les dades i iniciem sessió ens trobarem a la pantalla principal de la nostra aplicació.



Figura 20: Pantalla de log-in de l'aplicació.

---

### 13.1.2 Magatzem

Com podem veure ens trobem a la pantalla principal, on es pot visualitzar i moure'ns pel nostre magatzem desplaçant amb el dit, veiem tots els passadissos, i estanteries que tenim en el nostre magatzem. També podem observar que ens surt la icona del menú per accedir a les diferents opcions i facilitar-nos l'ús de l'aplicació.

En aquesta pantalla també podem registrar un paquet sense haver d'entrar el menú, simplement podem fer clic en una de les estanteries, i se'n obrirà la pantalla de selecció de prestatge a on guardarem el nostre paquet, d'aquesta manera accedim a registrar un paquet amb la posició on el guardarem ja seleccionada.

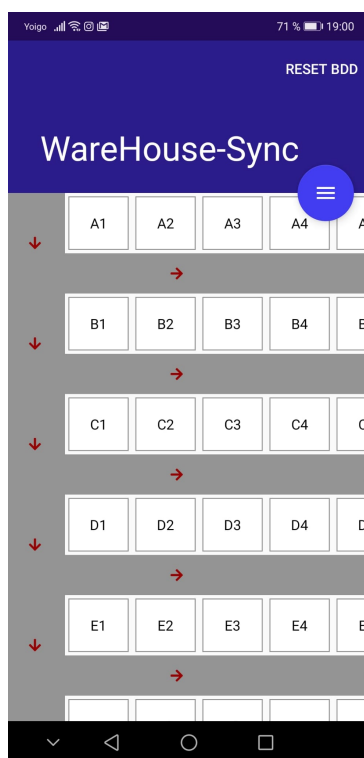


Figura 21: Pantalla principal on es veu el mapa del magatzem.

---

### 13.1.3 Menú de l'aplicació

Quan desplaguem el menú podem veure les diferents opcions que ens permet l'aplicació.

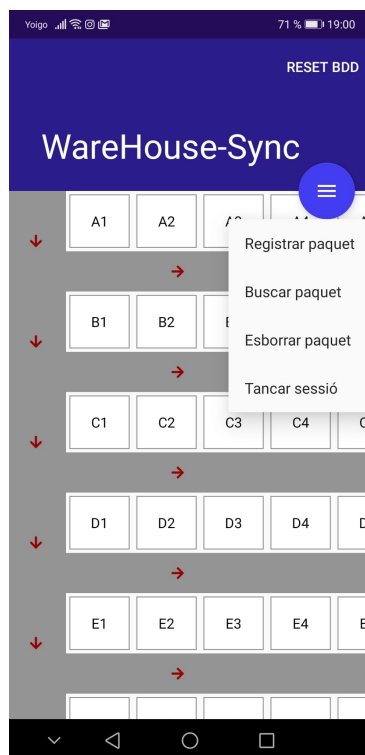


Figura 22: Pantalla de menú de WareHouse-Sync

1. **Registrar paquet** -> S'accedirà a la pantalla de registrar un paquet, sense cap localització d'on el volem guardar dins del magatzem. Ens demanarà les dades del paquet, i ens farà llegir el xip NFC del paquet, per tal de guardar-lo.
2. **Buscar paquet** -> Ens permet veure, buscar i seleccionar un paquet de la nostra BBDD, quan en seleccionem un ens mostrarà visualment a on es troba guardat, estanteria i prestatge, pintant la seva ubicació en el mapa del magatzem.
3. **Eliminar paquet** -> Ens portarà a una pantalla molt semblant a la de buscar un paquet, però ens permetrà treure'l del seu prestatge i eliminar-lo d'aquella localització, **Exemple:** Quan un paquet deixa d'estar guardat al magatzem i passa a ser enviat.
4. **Log out** -> Opció que ens permet tancar el nostre compte d'usuari, i ens porta a la pantalla del log in, per una futura connexió.

---

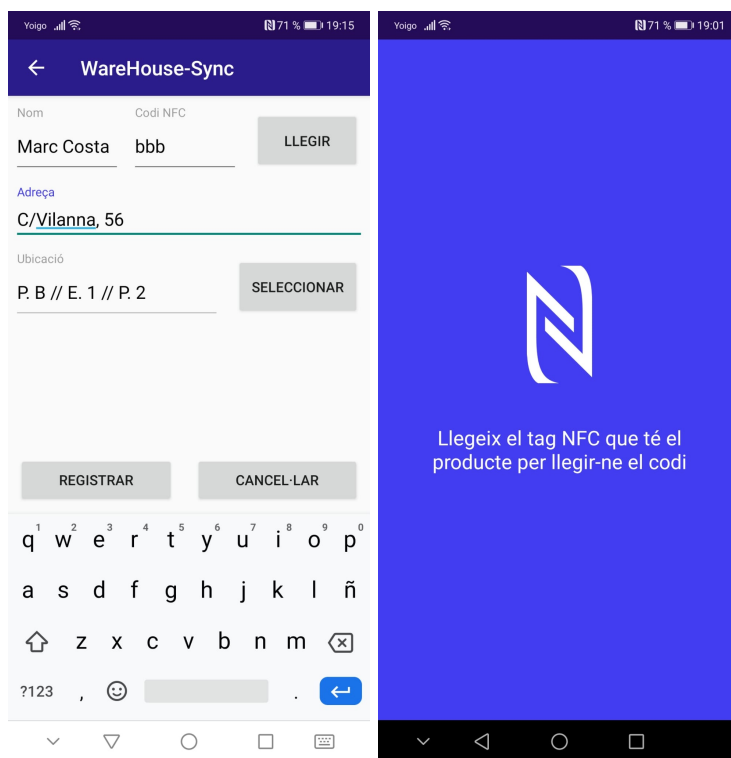
### 13.1.4 Registrar paquet

En aquesta opció pot entrar-hi de dues formes, la primera és seleccionant la ubicació en el mapa del magatzem i ja tenim la ubicació escollida. La segona és entrant des del menú desplegable i seleccionant Registrar paquet, en aquesta opció s'haurà de seleccionar l'ubicació, ja que no en tenim cap d'escollida, prement el botó seleccionar ens portarà el mapa del magatzem principal i ens deixarà seleccionar un prestatge.

En aquest formulari entrem les dades del paquet, per tal de registrar-lo a la nostra BBDD, s'ha d'introduir el Nom a qui va adreçat, l'adreça, i s'haurà de llegir el xip NFC incorporat en el paquet.

Per llegir el xip NFC simplement cliquem en L·Llegir i ens escanjarà el chip del paquet, quan sigui llegit ens tornarà a la pantalla del formulari.

Un cop entrades totes les dades, podem guardar el paquet en el nostre magatzem i a la nostra BBDD. En la pantalla b) ens trobem amb aquesta pantalla, l'aplicació està esperant la lectura d'un xip NFC del paquet, ja sigui pel formulari de registrar-lo o el d'eliminar-lo.



(a) Registrar paquet

(b) Escanejant chip NFC

---

### 13.1.5 Buscar paquet

En aquesta opció ens permet visualitzar tots els paquets que tenim registrats a la nostra BBDD, veient tota les dades de cada un. Un cop trobat el que buscàvem, si el seleccionem ens portarà a la pantalla del mapa del magatzem visualitzant la seva posició pintada en color verd, fent la seva localització en el magatzem molt més fàcil i àgil.

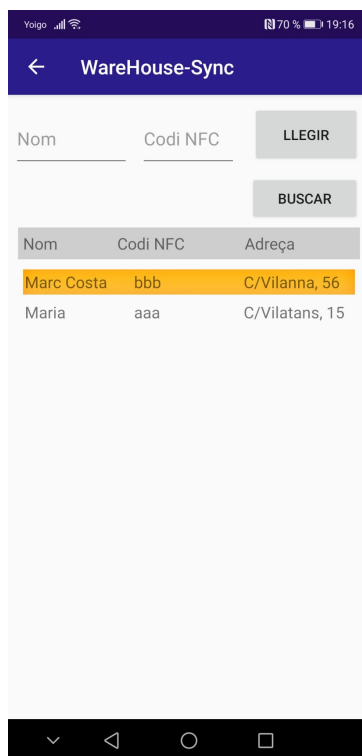


Figura 23: Vista de buscar i seleccionar un paquet.

---

### 13.1.6 Eliminar paquet

Ens permet treure un paquet del seu prestatge, la pantalla del formulari actua com la de "Buscar paquet", podem buscar-lo per nom o també es pot llegir amb el xip NFC, però un cop en seleccionem un podem esborrar-lo del mapa del magatzem, i de la nostra BBDD com a paquet emmagatzemat. **\*Cal vigilar de no esborrar un paquet erroni, ja que no es pot tornar enrere.**

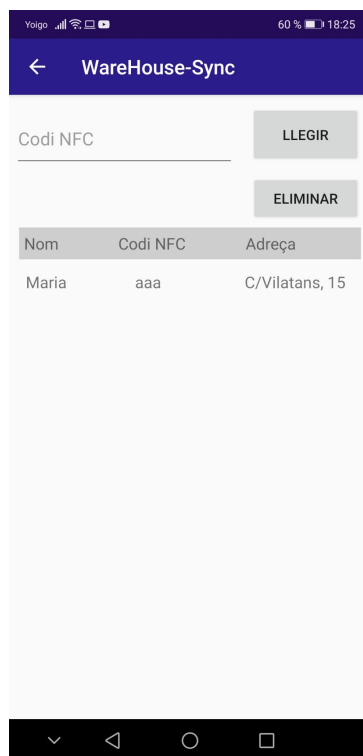
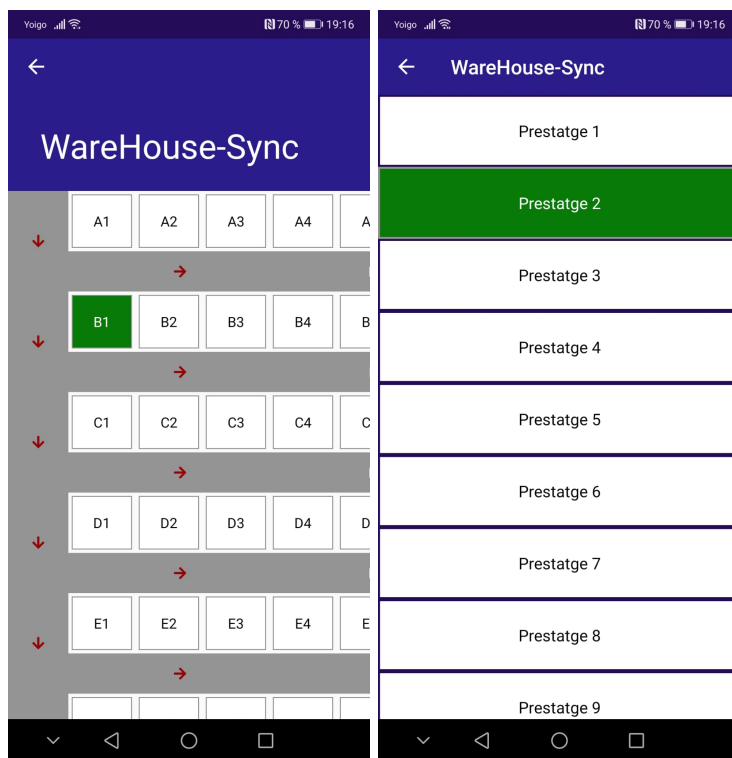


Figura 24: Pantalla per eliminar un paquet del magatzem.

### 13.1.7 Localització del paquet

Un cop seleccionat un paquet de l'opció "Buscar paquet", ens sortirà el mapa del magatzem de manera que ens marcarà la seva ubicació dins dels passadissos del magatzem, en aquesta vista només podrem accedir en l'estanteria i prestatge on conté el paquet.

Si accedim dins de l'estanteria on es troba el paquet, veiem a quin dels prestatges el podem trobar, molt intuïtiu i simple.



(a) Localització del paquet en el mapa (b) Localització en els prestatges

---

## Referències

- [1] <https://ecommercedb.com/>
- [2] <https://www.shopify.com/enterprise/global-ecommerce-statistics>
- [3] <https://www.statista.com/statistics>
- [4] <https://developer.android.com/training/data-storage/room>
- [5] <https://www.arduino.cc/en/Tutorial/BuiltInExamples>
- [6] <https://es.stackoverflow.com/>
- [7] <https://github.com/android/views-widgets-samples/tree/master/RecyclerView/#readme>
- [8] <https://www.zebra.com/es/es.html>
- [9] <https://developer.android.com/samples>
- [10] [https://www.tutorialspoint.com/android/android\\_activities.htm](https://www.tutorialspoint.com/android/android_activities.htm)
- [11] <https://www.arduino.cc/en/Tutorial/BuiltInExamples>
- [12] <https://developer.android.com/guide/topics/connectivity/nfc/nfc>