



Gestió d'una biblioteca amb Android

Andreu Querol Salsench

Grau d'Enginyeria Informàtica

Desenvolupament d'aplicacions per a dispositius mòbils (Android)

Consultors: Joan Vicent Orenge Serisuelo, Antonio Rodríguez Gutiérrez.

Professor responsable de l'assignatura: Carles Garrigues Olivella

06/2017



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

FITXA DEL TREBALL FINAL

Títol del treball:	<i>Gestió d'una Biblioteca amb Android</i>
Nom de l'autor:	<i>Andreu Querol Salsench</i>
Nom del consultor/a:	<i>Joan Vicent Orenge Serisuelo, Antonio Rodríguez Gutiérrez.</i>
Nom del PRA:	<i>Carles Garrigues Olivella</i>
Data de lliurament (mm/aaaa):	<i>06/2017</i>
Titulació o programa:	<i>Grau d'Enginyeria Informàtica</i>
Àrea del Treball Final:	<i>Desenvolupament d'aplicacions per a dispositius mòbils (Android)</i>
Idioma del treball:	<i>Català</i>
Paraules clau	<i>Android, Biblioteca, Gestió</i>
<p>Resum del Treball (màxim 250 paraules): <i>Amb la finalitat, context d'aplicació, metodologia, resultats i conclusions del treball</i></p>	
<p>La finalitat d'aquest projecte és la de proporcionar als usuaris d'una biblioteca una aplicació per a dispositius Android que els hi permeti accedir als serveis que ofereix la biblioteca, com ara préstec o reserva de llibres, de forma més ràpida i moderna.</p> <p>Avui en dia la gran majoria de la població, especialment els més joves, tenen a disposició un telèfon intel·ligent. Android és el sistema operatiu més utilitzat amb una gran diferència respecte als seus competidors, ja que en el 2016 aproximadament el 86% dels telèfons intel·ligents del món eren dispositius Android. És en aquest context en el qual aquest projecte obté el seu punt de partida i pretén oferir als usuaris de la biblioteca un sistema d'interacció més adaptat als nous temps i a la vegada voldria intentar captar una mica l'atenció de les noves generacions a les, de vegades oblidades, biblioteques.</p> <p>Com a resultat d'aquest projecte es vol aconseguir principalment dos objectius, per una banda que l'autor aprengui el màxim possible sobre la tecnologia Android, és per aquest motiu que es desenvoluparà una aplicació des de zero, sense aprofitar codi o dissenys ja existents en altres aplicacions similars. Per altra banda es vol desenvolupar una aplicació completa, simple de fer servir i lliure d'errors.</p>	

Abstract (in English, 250 words or less):

The purpose of this project is to provide the users of a library an application for Android devices that allows them to access the services offered by the library, such as loan or reserve books, in a faster and modern way.

Today the vast majority of the population, especially the younger ones, have at their disposal a smartphone. Android is the most used operating system with a big difference compared to its competitors, at 2016 approximately 86% of smart phones in the world were Android devices. It is in this context where this project gets its starting point and seeks to provide to the users of the library an interaction system adapted to the new times and also it tries to catch some attention of the new generations to the, sometimes forgotten, libraries.

As a result of this project we want to achieve two main objectives, firstly the author should be able to learn as much as possible about the Android technology, for that reason the application will be developed from scratch, without using existing code or designs from other similar applications. On the other hand we want to develop a complete application, simple to use and free of errors.

Index

1- Introducció	pàgina 3
1.1 - Context i justificació del treball	pàgina 3
1.2 - Objectius del Treball	pàgina 4
1.3 - Enfocament i mètode seguit	pàgina 5
1.4 - Planificació del Treball	pàgina 6
1.5 - Sumari de productes obtinguts	pàgina 7
1.6 - Descripció dels altres capítols de la memòria	pàgina 8
2 - Disseny centrat en l'usuari	pàgina 9
2.1 - Usuaris i context d'ús	pàgina 9
2.2 - Entrevistes en profunditat	pàgina 9
2.3 - Anàlisi competitiva	pàgina 10
2.4 - Resultats	pàgina 11
2.4.1 - Resultats de les entrevistes en profunditat	pàgina 11
2.4.2 - Resultats anàlisi competitiva	pàgina 13
2.5 - Conclusions	pàgina 14
2.5.1 - Conclusions entrevistes en profunditat	pàgina 14
2.5.2 - Conclusions Anàlisi competitiva	pàgina 15
2.6 - Perfil d'usuaris	pàgina 16
2.6.1 - Usuari habitual d'una biblioteca	pàgina 16
2.6.2 - Usuari no habitual d'una biblioteca	pàgina 17
2.7 - Escenaris	pàgina 18
2.8 - Flux d'interacció	pàgina 19
2.9 - Sketches	pàgina 20
2.10 - Prototipus d'alta fidelitat	pàgina 22
2.11 - Avaluació	pàgina 24
2.11.1 - Recull de preguntes	pàgina 24
2.11.2 - Tasques que haurien de realitzar els usuaris	pàgina 25
3 - Disseny tècnic	pàgina 28
3.1 - Diagrama de casos d'ús	pàgina 28
3.2 - Casos d'ús	pàgina 29
3.3 - Disseny de la base de dades	pàgina 32
3.3.1 - Diagrama UML	pàgina 32
3.3.2 - Model lògic relacional	pàgina 32
3.4 - Disseny de classes	pàgina 33
3.4.1 - Diagrama UML del servidor	pàgina 33
3.4.2 - Diagrama UML de l'aplicació Android	pàgina 34
3.5 - Disseny de l'arquitectura	pàgina 35
3.5.1 - Diagrama de l'arquitectura a alt nivell	pàgina 36
4 - Implementació	pàgina 37
4.1 - Descripció d'eines, editors i API utilitzats	pàgina 37
4.1.1 - Eines i editors	pàgina 37
4.1.2 - API utilitzades	pàgina 38
4.2 - Decisions preses	pàgina 40
4.2.1 - Implementació d'un servidor propi	pàgina 40
4.2.2 - Utilització de les API	pàgina 40
4.2.3 - Creació d'usuaris	pàgina 40

4.2.4- Ampliació de préstec de llibres	pàgina 41
4.2.5 - Categories de cerca	pàgina 41
4.2.6 - Aplicació bibliotecari	pàgina 41
4.2.7 - Retorn de llibres	pàgina 41
4.2.8 - Home sense menú	pàgina 42
4.3- Codi font Android	pàgina 42
4.3.1 - RecyclerView i Adaptadors	pàgina 42
4.3.2 - Detalls d'un llibre i invocació del servidor	pàgina 45
4.3.3 - Login / Logout i emmagatzemament de l'usuari al telèfon	pàgina 47
4.3.4 - Sol·licitar permisos per accedir a la càmera	pàgina 49
4.4- Codi font del Servidor Restful	pàgina 50
4.5- Anàlisi de l'estat del projecte	pàgina 52
5 - Proves	pàgina 53
5.1- Com es realitzaran les proves	pàgina 53
5.2 - Proves amb Espresso i JUnit	pàgina 53
5.3 - Proves amb l'emulador Android	pàgina 54
5.4 - Proves amb el telèfon físic	pàgina 57
6 - Conclusió	pàgina 59
7 - Bibliografia	pàgina 60
8 - Glossari	pàgina 61
9 - Annex	pàgina 62
9.1- Aplicació Bibliotecari	pàgina 62
9.2 - Comentaris.....	pàgina 66
9.2.1 - Usuaris per defecte.....	pàgina 66
9.2.2 - Codis QR vàlids.....	pàgina 66
9.2.3 - Informació dels llibres.....	pàgina 66

1 - Introducció

1.1 - Context i justificació del Treball

Durant aquest projecte es vol desenvolupar una aplicació de gestió per a una biblioteca en Android, amb aquesta aplicació el que es vol aconseguir és oferir un conjunt de funcionalitats als usuaris de la biblioteca per a que els hi sigui més fàcil i còmode aprofitar tots els serveis que la biblioteca ofereix.

Avui en dia tant les institucions com les empreses aposten cada vegada més pel mercat de les aplicacions mòbils, oferint als seus usuaris funcionalitats extra que els i poden donar un valor afegit a la seva marca o producte. En aquest context penso que és molt interessant que una biblioteca incorpori una aplicació per a dispositius mòbils, ja que per una banda permet fer la biblioteca més accessible als seus usuaris habituals, i per altra banda és una plataforma que pot animar a les noves generacions, més habituades a Internet que a les biblioteques, a interessar-se pels llibres i els serveis que ofereix.

És molt probable que arreu del món hi hagi múltiples biblioteques que ja ofereixin una aplicació d'aquest estil, concretament en l'àmbit Català podem trobar per exemple l'aplicació [BibliotequesXBM](#), desenvolupada per la diputació de Barcelona. El principal punt fort d'aquesta aplicació és que permet conèixer els esdeveniments que es duen a terme a les biblioteques de Barcelona i també permet saber quins són els autobusos més pròxims a l'usuari que pot utilitzar per arribar a una biblioteca en concret. Per altra banda el principal punt feble que destaquen els seus usuaris és la falta de fluïdesa i estabilitat.

En el cas d'aquest projecte es vol innovar respecte a les aplicacions ja existents a l'hora d'identificar cada un dels llibres de la biblioteca mitjançant un codi QR, d'aquesta forma els usuaris poden accedir a la informació dels llibres sense haver de buscar-los a cap llista o cercador.

1.2 - Objectius del Treball

L'objectiu del treball és desenvolupar una aplicació amb Android que faciliti l'ús dels serveis d'una biblioteca als seus usuaris, concretament els requisits a alt nivell són els següents:

Requisits funcionals:

- Permetre buscar quins llibres hi ha d'una certa temàtica.
- Disposar d'un sistema per a poder reserva llibres de la biblioteca.
- Que els usuaris puguin consultar quins llibres tenen reservats o si un llibre en concret ja està reservat per algun altre usuari.
- Informar a l'usuari si ha de retornar algun llibre pròximament.
- Permetre ampliar el temps de préstec d'un llibre si no hi ha una altra reserva.
- Poder gestionar l'alta / modificació / baixa d'usuaris.
- Poder gestionar l'alta / modificació / baixa de llibres.
- Disposar de la funcionalitat necessària per a prestar un llibre a un usuari.
- Permetre la identificació dels llibres mitjançant un codi QR.

Requisits no funcionals:

- Ser senzilla de fer servir, pràctica i moderna.
- Estar desenvolupada per a dispositius Android.

Requisits opcionals que poden quedar fora de l'abast del projecte:

- Permetre la participació en enquestes per a votar quins són els pròxims llibres que hauria d'adquirir la biblioteca.
- Permetre la reserva de sales d'estudi o d'ordinadors durant unes certes hores.
- Incorporar un servei de préstec de joguines per a nens petits.
- Incorporar un servei de préstec de pel·lícules o diaris.

Personalment el meu principal objectiu és el d'aprendre tant com es pugui la tecnologia Android, ja que considero que per al meu futur professional pot ser molt interessant tenir una mica d'experiència en aquesta àrea.

1.3 - Enfocament i mètode seguit

Avui en dia ja existeixen aplicacions similars a la que es vol desenvolupar, no obstant he decidit no aprofitar cap codi o disseny ja existeix i desenvolupar un producte nou des de zero, ja que en ser el primer cop que desenvolupo una aplicació en Android vull poder aprofitar el projecte per obtenir l'experiència d'implementar una aplicació completament nova.

L'estratègia a seguir per a dur a terme el projecte consisteix en dividir-lo en diferents tasques d'alt nivell:

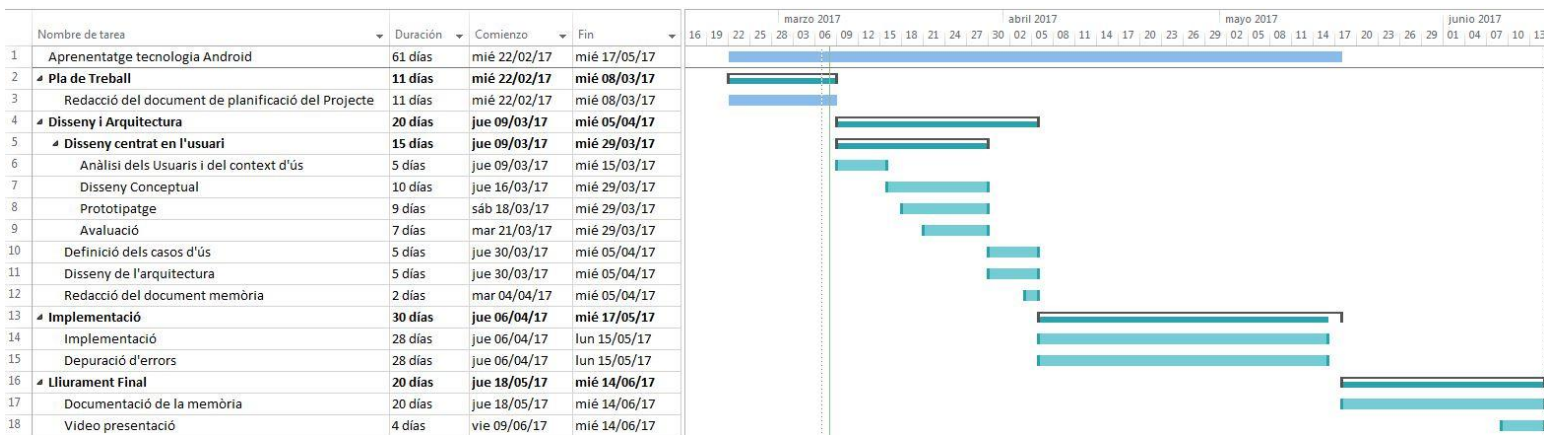
- *Aprenentatge de la tecnologia Android:*
Considero aquesta tasca com un dels reptes principals del projecte, ja que Android és una tecnologia completament nova per a mi. Aquesta tasca es durà a terme de forma paral·lela a la resta de tasques durant gairebé tot el cicle de vida del projecte.
- *Planificació del projecte:*
En aquesta tasca es planifica com abordar el projecte, definint quins són els objectius i l'abast del projecte, així com la planificació temporal i l'estratègia que es vol seguir per dur-lo a terme.
- *Estudi dels usuaris i disseny de l'aplicació:*
En aquesta tasca s'estudiarà quines són les necessitats a cobrir dels usuaris i es dissenyaran tots els artefactes que compondran l'aplicació per tal de poder resoldre aquestes necessitats, des de la interfície gràfica, passant per el propi disseny intern de l'aplicació fins al disseny de la base de dades.
- *Implementació:*
En aquesta tasca s'implementarà el disseny realitzat en la tasca anterior.
- *Depuració d'errors i problemes:*
En aquesta tasca es buscaran i corregiran els errors que sorgeixin durant la implementació.
Amb la finalitat de detectar i corregir els errors tan aviat com sigui possible, aquesta tasca i la tasca anterior d'implementació es duran a terme en paral·lel, d'aquesta forma es podrà evitar que els errors puguin arribar a expandir-se a altres parts de l'aplicació.
- *Lliurament final:*
En aquesta taca es preparà el lliurament definitiu del projecte, realitzant la presentació virtual i la posterior defensa davant del tribunal d'avaluació.

1.4 - Planificació del Treball

Per a la realització del projecte es dedicaran de 2 a 3 hores diàries tant en dies laborables com festius. Les dates clau per a la realització del projecte són les següents:

Tasca	Data d'entrega
Pla de treball (PAC 1)	08/03/2017
Disseny i arquitectura (PAC 2)	05/04/2017
Implementació (PAC 3)	17/05/2017
Lliurament final	14/06/2017

En el diagrama de Gantt següent es pot veure la planificació temporal que es durà a terme:



En el diagrama anterior es mostren els dies naturals que s'invertiran en cada tasca, en la següent taula es pot veure la relació d'hores que es dedicaran a cada una de les tasques:

PAC	Tasca	Hores
0- Aprenentatge	Aprenentatge tecnologia Android	66
1- Pla de treball	Redacció del document planificació del projecte	10
2- Disseny i arquitectura	Anàlisi dels usuaris i del context d'ús	10
	Disseny conceptual	9
	Prototipatge	7
	Avaluació	4
	Definició dels casos d'ús	5
	Disseny de l'arquitectura	7
	Redacció del document memòria	2
3- Implementació	Implementació	40
	Depuració d'errors	20
4- Lliurament final	Documentació de la memòria	20
	Vídeo presentació	8

**En la planificació exposada anteriorment es va subestimar el que costaria implementar l'aplicació, especialment la part de desenvolupar el servidor, per tal de corregir aquest fet durant l'etapa d'implementació es va dedicar l'hora extra que es tenia reservada cada dia per aprendre Android a programar l'aplicació, ja que al cap i a la fi amb la pràctica de programar també es seguia aprenent Android, per tant es van sumar unes 30 hores a la implementació, restant-les de l'aprenentatge d'Android.

A més com es va anar depurant els errors a mesura que s'anava desenvolupant l'aplicació, es podria considerar que la majoria d'hores inicialment pensades en depurar errors també es van acabar invertint en implementar l'aplicació per tant la majoria d'aquestes hores també s'han acabat dedicant a implementar el projecte.

Finalment per tal de donar els últims retocs a l'aplicació es va dedicar 20 hores extra les quals es van extreure del temps planificat per a la redacció de la memòria final.

Tenint en compte aquestes desviacions la relació d'hores reals que s'ha invertit en el projecte han estat les següents:

PAC	Tasca	Hores
0- Aprenentatge	Aprenentatge tecnologia Android	36
1- Pla de treball	Redacció del document planificació del projecte	10
2- Disseny i arquitectura	Anàlisi dels usuaris i del context d'ús	10
	Disseny conceptual	9
	Prototipatge	7
	Avaluació	4
	Definició dels casos d'ús	5
	Disseny de l'arquitectura	7
	Redacció del document memòria	2
3- Implementació	Implementació	80
	Depuració d'errors	10
4- Lliurament final	Documentació de la memòria	20
	Vídeo presentació	8
TOTAL		208

1.5 - Sumari de productes obtinguts

En finalitzar el projecte s'han creat els següents productes:

- Document memòria del projecte.
- Vídeo presentació del projecte.
- APK de l'aplicació Biblioteca (Usuari) desenvolupada juntament amb el codi font.
- APK de l'aplicació Bibliotecari desenvolupada juntament amb el codi font.
- Codi font del servidor RestFul desenvolupat.
- Codi SQL corresponent a la base de dades.

1.6 - Descripció dels altres capítols de la memòria

- *Estudi dels usuaris:*
Es realitza un estudi de quines son les necessitats dels usuaris que s'han de cobrir.
- *Disseny i arquitectura:*
Es detalla el disseny que s'utilitza en tots els artefactes que componen el projecte.
- *Implementació:*
Es detalla els reptes superats durant la codificació del projecte així com altra informació rellevant relativa a la implementació d'aquest.
- *Proves realitzades:*
Es detallen les proves que s'han dut a terme per a garantir que el projecte s'ha dut a terme amb èxit.
- *Conclusió:*
Es fa un breu resum dels aspectes més rellevants del projecte amb la finalitat de donar-li una conclusió.
- *Bibliografia:*
S'inclouen totes les referències bibliogràfiques utilitzades en el desenvolupament del projecte.
- *Glossari:*
S'inclou una petita descripció de termes tècnics utilitzats en la memòria.

2 - Disseny centrat en l'usuari

2.1 - Usuaris i context d'ús

Per a realitzar la indagació dels usuaris en aquest projecte s'ha escollit utilitzar els mètodes d'anàlisi competitiva i entrevistes en profunditat. El principal motiu per descartar els altres mètodes d'indagació és les restriccions de temps que hi ha per desenvolupar aquesta fase en el projecte i també la falta d'un usuari / grup d'usuaris propers al meu entorn que estiguin habituats a utilitzar una biblioteca i estiguin disposats a prestar-se a ser subjectes d'estudi.

No obstant considero que els mètodes escollits donaran una base prou sòlida per a descobrir com els possibles usuaris de l'aplicació interactuen amb el seu telèfon intel·ligent tot i descobrint quines són les funcionalitats que més els hi podrien interessar de l'aplicació que s'està desenvolupant.

A l'hora de realitzar les entrevistes amb profunditat s'han confeccionat un formulari amb vint-i-dues preguntes, algunes de les preguntes es poden respondre amb: Si / No / No ho sé. D'altres són preguntes amb respostes més obertes amb les quals s'espera poder conèixer els hàbits i els problemes dels usuaris a l'hora d'interactuar amb les biblioteques i d'aquesta forma descobrir alguna dificultat que l'aplicació els hi pugui resoldre.

2.2 - Entrevistes en profunditat

Les preguntes que s'han utilitzat en l'entrevista són les següents:

Sexe:

Edat:

Pregunta	Respostes		
Disposa d'un telèfon intel·ligent?	Si	No	No ho sé
És un telèfon amb sistema operatiu Android?	Si	No	No ho sé
Aproximadament, quantes hores al dia utilitza el telèfon?	Número d'hores		
Fa d'altres coses mentre utilitza el telèfon?	Si	No	
Te alguna dificultat a l'hora d'utilitzar el seu telèfon intel·ligent?	Si	No	
En cas afirmatiu, quina dificultat té?	Resposta oberta		
Li agrada provar noves aplicacions?	Si	No	
Presta atenció a la música o sons d'una aplicació? (sense tenir en compte aplicacions per a escoltar música o jocs)	Si	No	
Que és el que més valora quan es descarrega una nova aplicació?	Resposta oberta		
Algun cop ha instal·lat una nova aplicació i l'ha eliminat el mateix dia?	Si	No	
En cas afirmatiu, per quin motiu?	Resposta oberta		

Li agrada la lectura?	Si	No
Sol anar habitualment a una biblioteca?	Si	No
En cas negatiu, perquè no hi va?	Resposta oberta	
Estan en una biblioteca, algun cop ha tingut problemes per a trobar algun llibre que estes buscant?	Si	No
Per què?	Resposta oberta	
Algun cop ha utilitzat el servei de préstec de llibres d'una biblioteca?	Si	No
Ha tingut mai algun problema a l'utilitzar el servei de préstec de llibres?	Si	No
Algun cop ha tingut un llibre en préstec i ha oblidat el dia en què l'havia de retornar?	Si	No
Li pareix interessant l'aplicació que s'està desenvolupant en aquest projecte? (<i>en cas de tenir qualsevol dubte respecte a l'aplicació pot preguntar el que vulgui</i>)	Si	No
Troba a faltar algun servei que l'aplicació podria oferir i no ofereix?	Resposta oberta	
Altres comentaris, valoracions i opinions:	Resposta oberta	

2.3 - Anàlisi competitiva

Per a realitzar l'anàlisi competitiva s'ha escollit dues aplicacions, BibliosBCN i BibliotequesXBM, les dues són aplicacions que estan desenvolupades per a biblioteques de la zona de Barcelona. S'ha estudiat quines són les funcionalitats que ofereixen cada una i quines són les opinions que els seus usuaris tenen d'elles. A més s'han buscat els seus punts forts i punts febles per tal de tenir-los en compte i s'ha procurat no caure en els mateixos errors.

S'ha de dir que l'aplicació BibliosBCN és bastant diferent de l'aplicació que es vol desenvolupar en aquest projecte, no obstant s'ha decidit incloure-la en l'anàlisi competitiva, ja que considero que en ser una aplicació que ofereix funcionalitats per a les biblioteques sempre es pot extreure algun resultat útil del seu anàlisi.

2.4 - Resultats

2.4.1 - Resultats de les entrevistes en profunditat:

S'ha dut a terme l'entrevista a dotze usuaris propers al meu entorn, sense discriminar edat ni coneixements previs en tecnologia mòbil.

Pregunta	Respostes
Disposa d'un telèfon intel·ligent?	Si: 100% d'usuaris
És un telèfon amb sistema operatiu Android?	Si : 10 usuaris No: 2 usuaris
Aproximadament, quantes hores al dia utilitza el telèfon?	1 hora: 2 usuaris 2 hores: 3 usuaris 4 hores: 2 usuaris 5 hores: 2 usuaris 6 hores : 1 usuari 8 hores : 1 usuari 10 hores: 1 usuari
Fa d'altres coses mentre utilitza el telèfon?	Si : 8 usuaris No: 4 usuaris
Te alguna dificultat a l'hora d'utilitzar el seu telèfon intel·ligent?	No: 100% d'usuaris
En cas afirmatiu, quina dificultat té?	-
Li agrada provar noves aplicacions?	Si: 8 usuaris No: 4 usuaris
Presta atenció a la música o sons d'una aplicació? (sense tenir en compte aplicacions per a escoltar música o jocs)	Si: 3 usuaris No: 9 usuaris
Que és el que més valora quan es descarrega una nova aplicació?	-
Algun cop ha instal·lat una nova aplicació i l'ha eliminat el mateix dia?	Si: 9 usuaris No: 3 usuaris
En cas afirmatiu, per quin motiu?	-
Li agrada la lectura?	Si : 10 usuaris No: 2 usuaris
Sol anar habitualment a una biblioteca?	Si: 4 usuaris No: 8 usuaris
En cas negatiu, perquè no hi va?	-
Estan en una biblioteca, algun cop ha tingut problemes per a trobar algun llibre que estes buscant?	Si: 10 usuaris No: 2 usuaris
Per què?	-
Algun cop ha utilitzat el servei de préstec de llibres d'una biblioteca?	Si: 100% dels usuaris

Ha tingut mai algun problema a l'utilitzar el servei de préstec de llibres?	No: 100% dels usuaris
Algun cop ha tingut un llibre en préstec i ha oblidat el dia en què l'havia de retornar?	Si: 8 usuaris No: 4 usuaris
Li pareix interessant l'aplicació que s'està desenvolupant en aquest projecte? (en cas de tenir qualsevol dubte respecte a l'aplicació pot preguntar el que vulgui)	Si: 11 usuaris No: 1 usuari
Troba a faltar algun servei que l'aplicació podria oferir i no ofereix?	-
Altres comentaris, valoracions i opinions:	-

En les preguntes obertes els entrevistats han respost el següent:

- Que és el que més valora quan es descarrega una nova aplicació?*

La majoria d'usuaris valoren per davant de tot la facilitat d'ús i que tingui les funcionalitats que necessiten, altres aspectes que valoren són el disseny, la velocitat i la novetat.
- Algun cop ha instal·lat una nova aplicació i l'ha eliminat el mateix dia? En cas afirmatiu, per quin motiu?*

Nou dels dotze usuaris si han eliminat una aplicació el mateix dia d'instal·lar-la. El principal i gairebé únic motiu, ha sigut que l'aplicació no era el que esperaven i no complia amb les seves expectatives. L'altre motiu que ha portat a algun usuari a desinstal·lar una aplicació ha sigut l'excés de publicitat.
- Sol anar habitualment a una biblioteca? En cas negatiu, perquè no hi va?*

Vuit dels dotze usuaris no van habitualment a una biblioteca, en aquest cas hi ha dos motius principals, per una banda hi ha un grup d'usuaris que diuen que la informació la busquen a través d'Internet i per tant no tenen el costum d'anar a una biblioteca a buscar informació. Per altra banda hi ha també un altre grup d'usuaris que no van a la biblioteca per falta de temps.
- Estan en una biblioteca, algun cop ha tingut problemes per a trobar algun llibre que estes buscant? Per què?*

Deu dels dotze usuaris han tingut problemes a l'hora de trobar un llibre. La majoria de respostes fan referència al fet que no entenen l'organització o el sistema de classificació que es feia servir a la biblioteca. D'altres parlaven de la falta de continguts i sobretot de la falta d'actualització, especialment en llibres tècnics els quals passats uns anys poden perdre molta utilitat. Finalment un altre problema ha sigut a causa de que algú ha retornat el llibre i l'ha deixat en algun lloc que no corresponia.

- *Troba a faltar algun servei que l'aplicació podria oferir i no ofereix?*

En aquest cas les respostes han estat bastant diverses, cinc usuaris han respost que no troben a faltar cap servei més. No obstant altres usuaris si han proposat un conjunt d'idees força interessants:

- Tres usuaris voldrien disposar d'un servei de préstec de llibres electrònics amb l'aplicació, a poder ser voldrien poder llegir els llibres digitals sense connexió a Internet.
- Dos usuaris voldrien poder proposar a la biblioteca quins llibres haurien d'adquirir pròximament.
- Un usuari voldria poder reservar llibres utilitzant una llista d'espera en cas que el llibre ja estigues reservat.
- Un usuari voldria que se l'informes de quines noves obres va adquirint la biblioteca al llarg del temps, a més també voldria rebre informació sobre els esdeveniments o tallers que es realitzen a les biblioteques.

- *Altres comentaris, valoracions i opinions:*

Els usuaris han aportat diversos comentaris bastant interessants, en resum fan referència al fet que el projecte els hi pareix una bona idea, ja que podria estalviar temps als usuaris de les biblioteques. Per altra banda també han comentat que és una bona forma d'atansar la biblioteca als usuaris que estan acostumats a realitzar totes les seves accions per Internet, tal com van fer en el seu moment per exemple els bancs o les tendes.

2.4.2 - Resultats anàlisi competitiva:

Aplicació [BibliosBCN](#):

Funcionalitats destacades:

- Consultar les activitats que es duen a terme a les biblioteques de Barcelona.
- Consultar quina és la biblioteca que està més propera.
- Opció per a compartir les activitats als teus contactes.
- Calendari per a consultar les activitats que es duen a terme en un dia en concret.

Punts forts:

- Disposa de funcionalitats interessants, com per exemple poder conèixer quina és la biblioteca que està més propera a l'usuari.
- L'aplicació és bastant ràpida.

Punts febles:

- Es podria millorar el disseny de l'aplicació.

- La usabilitat no és gaire bona, no hi ha un buscador de biblioteques i per saber els horaris d'una biblioteca primer s'ha de buscar entre la llista d'esdeveniments.
- Pot proporcionar informació errònia, per exemple la funcionalitat per a mostrar els actes que es duen a terme prop de l'usuari no funciona bé.

Aplicació [BibliotequesXBM](#):

Funcionalitats destacades:

- Consultar les activitats que es duen a terme a la xarxa de biblioteques municipals i als bibliobusos.
- Consultar quina és la biblioteca que està més propera.
- Buscador avançat per a cercar actes i esdeveniments a les biblioteques.
- Permet consultar el catàleg de llibres i fer-ne una reserva o cancel·lar-la.
- L'usuari pot veure quins llibres té en préstec i els pot renovar.
- Identificació dels usuaris mitjançant un carnet amb codi de barres.

Punts forts:

- Disposa de moltes funcionalitats, és molt completa.
- Està disponible en Català, Castellà i Anglès.

Punts febles:

- L'aplicació és una mica inestable, per exemple si es fan més de dues consultes consecutives al catàleg apareix un error i es bloqueja el catàleg.
- El cercador del catàleg és poc usable, no té categories i pot ser molt difícil trobar un llibre d'una temàtica en particular a no ser que es conegui el títol o l'autor d'algun llibre en concret.
- Algunes pantalles de l'aplicació presenten un disseny poc depurat.

2.5 - Conclusions

2.5.1 - Conclusions entrevistes en profunditat:

- Els usuaris no tenen dificultat a l'hora d'utilitzar els seus telèfons, sincerament esperava que algun tingues alguna dificultat, especialment visual, ja que les pantalles solen ser relativament petites.
- A la majoria d'usuaris els hi agrada provar noves aplicacions, per tant no tindrien cap problema en provar la que s'està desenvolupant en aquest projecte.

- La majoria d'usuaris no presten atenció als sons d'una aplicació si no és de jocs o de reproducció de música, per tant considero que no val la pena incloure àudio en l'aplicació d'aquest projecte.
- Tenint en compte les opinions dels usuaris sobre que més valoren en descarregar una nova aplicació, i perquè han arribat a desinstal·lar una aplicació el mateix dia d'instal·lar-la, s'intentarà maximitzar l'esforç en crear una aplicació fàcil d'utilitzar i amb les funcionalitats necessàries per a que sigui útil als usuaris.
- La majoria d'usuaris han oblidat en alguna ocasió que havien de retornar un llibre, l'aplicació pot ser molt útil per a recordar als usuaris quan l'han de retornar.
- La majoria d'usuaris no solen anar habitualment a una biblioteca, crec que aquesta aplicació podria ajudar que hi anessin més sovint, ja que per exemple podria connectar la biblioteca amb els usuaris que estan més habituats a buscar la informació per Internet.
- La majoria d'usuaris han tingut problemes a l'hora de trobar un llibre a una biblioteca, en aquest cas l'aplicació podria facilitar la cerca de llibres als usuaris, evitant així els problemes que han tingut per no entendre el sistema de classificació de la biblioteca o perquè buscaven un llibre que ja havia estat prestat a una altra persona.
- Per altra banda un cop trobat el llibre els usuaris no han tingut mai cap problema amb el servei de préstec de llibres en si, per tant és un servei que l'aplicació no caldria que intentés millorar, ja que ja funciona correctament.
- Finalment els usuaris han suggerit moltes propostes bastant interessants que es podrien incorporar al projecte, no obstant a causa de l'abast que s'ha definit en el projecte penso que el millor és no incloure aquestes idees, almenys de moment, i deixar-les per possibles versions futures de l'aplicació.

2.5.2 - Conclusions Anàlisi competitiva:

Les dues aplicacions estudiades presenten uns punts forts i unes funcionalitats molt interessants, per exemple poder consultar quina és la biblioteca que està més propera i les activitats que s'hi duen a terme són funcionalitats que en un futur es podrien incorporar a l'aplicació perfectament, no obstant igual que en el cas anterior a causa de l'abast ja definit s'hauran de quedar fora del projecte. Per altra banda estudiant els punts febles de les aplicacions es pot arribar a la conclusió que per tal de millorar les aplicacions ja existents s'hauria d'oferir una interfície gràfica més depurada i fàcil d'utilitzar, amb un buscador de llibres millorat i que no presentes errors.

2.6 - Perfil d'usuaris

Considero que els perfils d'usuari més rellevants per a l'aplicació que s'està desenvolupant serien els següents:

2.6.1 - Usuari habitual d'una biblioteca.

Característiques del perfil:

Demografia	Adult, entre 40 i 60 anys.
Interessos i motivacions	La seva principal motivació per utilitzar l'aplicació consistirà en estalviar-se temps, ja que l'aplicació permetrà a l'usuari buscar els llibres que li interessin en qualsevol moment i saber si estarà disponible en el moment que vulgui visitar la biblioteca.
Experiència amb l'ús de la tecnologia mòbil	Relativament baixa, utilitza el seu telèfon intel·ligent entre 1 i 2 hores al dia.
Realitza altres tasques mentre utilitza el telèfon mòbil	No

Context d'ús:

Aquest perfil d'usuari no realitza altres tasques mentre utilitza el seu telèfon intel·ligent, per tant és molt probable que utilitzi l'aplicació en qualsevol moment mentre està a casa, prestant tota l'atenció a l'aplicació i sense gaires distraccions. Considero que utilitzarà l'aplicació des de casa, ja que a fora de casa segurament estaria duent a terme d'altres activitats i per tant no utilitzaria el telèfon mòbil.

Tasques:

- Buscar llibres al catàleg de la biblioteca.
- Reservar llibres del catàleg.
- Consultar si un llibre ja està reservat i quan es retornarà.
- Consultar quan ha de retornar un llibre que té en préstec.
- Ampliar el temps de préstec d'un llibre.
- Obtenir informació d'un llibre mitjançant el codi QR.

Llistat de característiques:

S'ha de mostrar clarament si el llibre que vol reservar l'usuari ja està en préstec, i en cas que ja estigui en préstec oferir l'opció de reservar-lo per a poder recollir-lo quan sigui retornat, a més també s'haurà de mostrar clarament en quina data estarà disponible el llibre. A banda el cercador de llibres ha de permetre filtrar els llibres per temàtiques, presentant aquests filtres de forma intuïtiva i natural. Finalment l'aplicació haurà de ser senzilla i fàcil de fer servir.

2.6.2 - Usuari no habitual d'una biblioteca

Característiques del perfil

Demografia	Adult, de totes les edats (entre 20 i 60 anys)
Interessos i motivacions	Aquest perfil d'usuari està acostumat a cercar la informació que necessita a través d'Internet mitjançant buscadors com Google. El principal interès que tindrà aquest perfil d'usuari en utilitzar l'aplicació serà obtenir una font més d'informació a banda de les que ja utilitza habitualment.
Experiència amb l'ús de la tecnologia mòbil	Relativament alta, de mitjana utilitza el seu telèfon intel·ligent unes 6 hores al dia.
Realitza altres tasques mentre utilitza el telèfon mòbil	Si

Context d'ús:

Aquest perfil d'usuari està acostumat a realitzar d'altres tasques mentre utilitza el seu telèfon intel·ligent, per tant prestarà una atenció parcial al telèfon mentre utilitza l'aplicació. A l'utilitzar el telèfon durant tantes hores al dia és molt probable que el faci servir tant des de casa com a fora de casa.

Tasques:

- Buscar llibres al catàleg de la biblioteca.
- Reservar llibres del catàleg.
- Consultar si un llibre ja està reservat i quan es retornarà.
- Consultar quan ha de retornar un llibre que té en préstec.
- Ampliar el temps de préstec d'un llibre.
- Obtenir informació d'un llibre mitjançant el codi QR.

Llistat de característiques:

S'ha de mostrar clarament si el llibre que vol reservar l'usuari ja està en préstec, i en cas que ja estigui en préstec oferir l'opció de reservar-lo per a poder recollir-lo quan sigui retornat, a més també s'haurà de mostrar clarament en quina data estarà disponible el llibre. A banda el cercador de llibres ha de permetre filtrar els llibres per temàtiques, presentant aquests filtres de forma intuïtiva i natural. Finalment l'aplicació haurà de ser senzilla i fàcil de fer servir.

2.7 - Escenaris

Escenari 1:

Perfil: Usuari habitual d'una biblioteca.

El Jaume és un home de 50 anys que després d'un dia de feina es relaxa a casa seva llegint un llibre d'una nova saga que està agradant-li molt. Mentre llegeix el llibre s'adona que està a punt d'acabar-lo i no té el llibre que continua amb la saga, aleshores decideix deixar un moment el llibre a una banda i agafa el seu telèfon mòbil. Obre l'aplicació i utilitza el cercador per a buscar al catàleg de la biblioteca el següent llibre de la saga introduint el títol del llibre, veu que la biblioteca té el llibre però ja ha estat reservat a una altra persona, no obstant també observa com el llibre serà retornat en tres dies. Per tant decideix apuntar-se a la cua de reserves del llibre per tal d'anar-lo a recollir quan ja estigui disponible, tot seguit tanca l'aplicació, deixa el telèfon i segueix llegint el llibre que havia deixat de banda.

Escenari 2:

Perfil: Usuari habitual d'una biblioteca.

La Paquita és una dona de 45 anys, es troba asseguda al sofà de casa seva llegint un llibre que ha agafat prestat de la biblioteca. Mentre llegeix, pensa que el llibre està agradant-li molt, però encara va per la meitat i demà l'haurà de tornar. De sobte recorda que fa uns dies es va instal·lar l'aplicació de la biblioteca al seu telèfon, així que l'obre i escaneja el codi QR que es troba al lloc del llibre, a continuació decideix ampliar el seu termini de reserva tres dies més, així s'assegura que el podrà acabar sense problemes. Un cop realitzada l'ampliació guarda el seu telèfon i continua llegint el llibre.

Escenari 3:

Perfil: Usuari no habitual d'una biblioteca.

La Sabi és una jove de 28 anys, havia sortit a passejar el seu gos i just en tornar a casa veu sobre el moble de l'entrada un llibre que havia agafat en préstec de la biblioteca. De cop recorda que ja fa dies que havia agafat el llibre i havia de tornar-lo, però no recorda quin dia ho ha de fer ni tampoc si ja s'ha passat del termini. Ràpidament obre l'aplicació amb el seu telèfon mòbil, que ja tenia a la mà perquè l'havia estat utilitzant mentre passejava, i escaneja el codi QR que té el llibre al seu lloc, ràpidament l'aplicació obre la informació referent al seu préstec del llibre, i la Sabi observa amb alleujament que encara té fins dimecres que ve per a retornar el llibre, tot seguit indica a l'aplicació que vol que el mateix dimecres li recordi que ha de retornar el llibre. A continuació tanca l'aplicació, va al menjador i es fica a mirar la televisió mentre parla per telèfon amb una amiga.

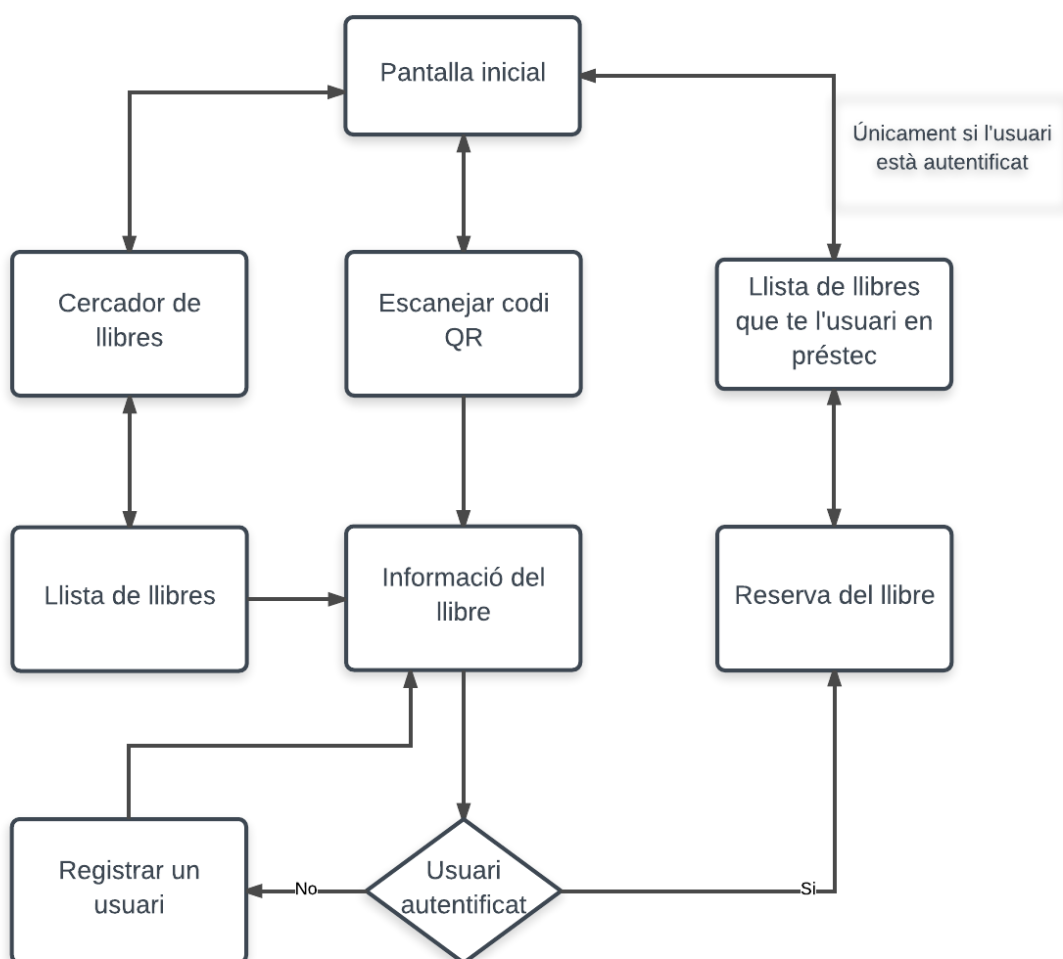
Escenari 4:

Perfil: Usuari no habitual d'una biblioteca.

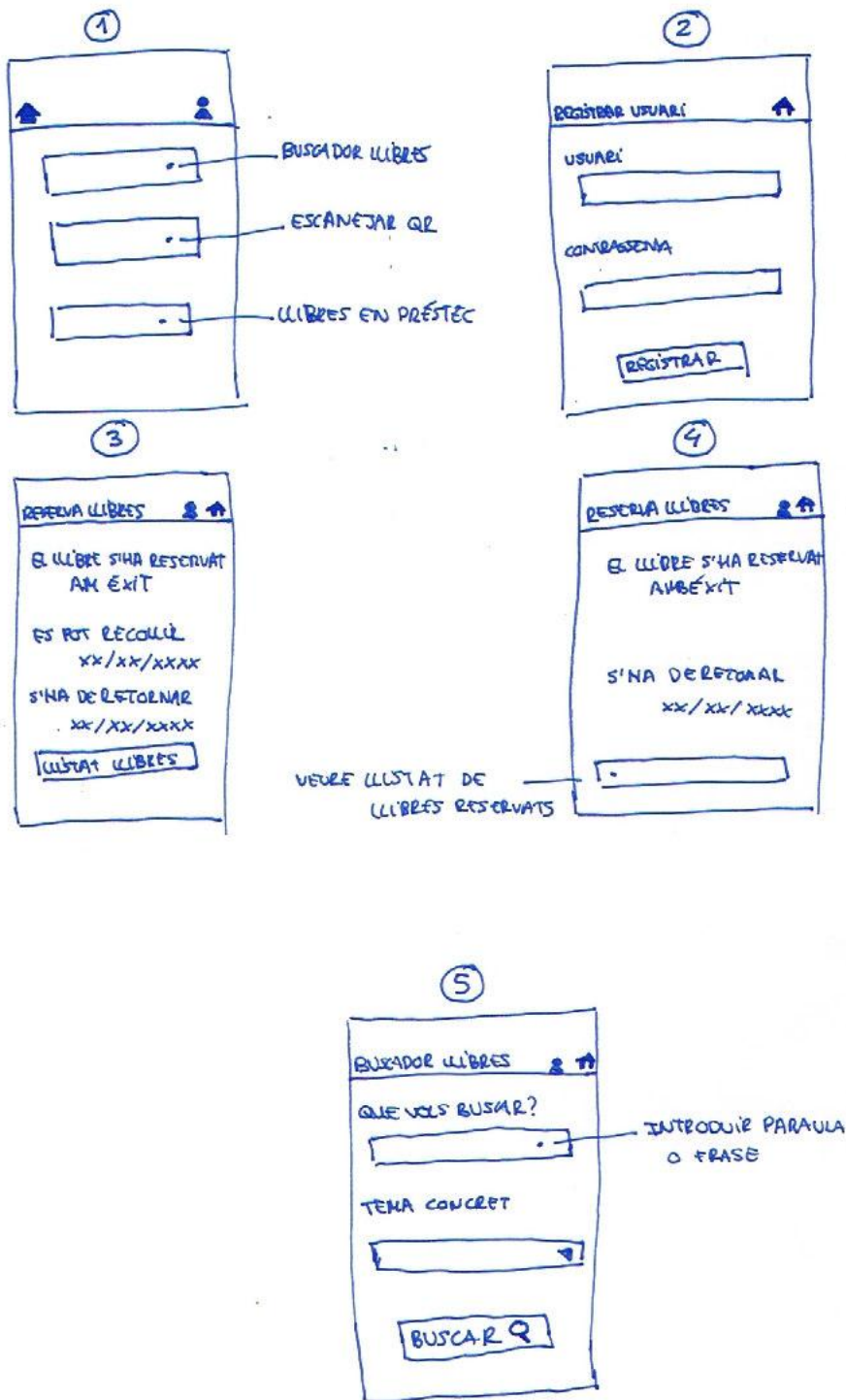
El Cristian és un jove de 27 anys, es troba a casa intentant resoldre uns problemes de física bastant complicats que ha d'entregar al seu professor de la universitat en dos dies i ja porta una bona estona buscant per Internet algun exemple resolt de problemes similars per intentar resoldre els que té més encallats, no obstant al ser un tema tan específic està costant-li molt trobar informació fiable i de qualitat. Al no trobar la informació que necessita es descarrega l'aplicació de la biblioteca que té al costat de casa i immediatament utilitza el cercador amb la funcionalitat de buscar per temes, d'aquesta forma espera poder trobar algun llibre de física que el pugui ajudar amb els seus dubtes. Donant un cop d'ull a la llista de llibres que li ofereix l'aplicació en troba un que pot ser interessant, el selecciona i llegeix tota la informació referent al llibre. Creu que finalment ha trobat la informació que necessita i com el llibre no està en préstec i té la biblioteca al costat de casa decideix tancar l'aplicació i sortir a buscar-lo.

2.8 - Flux d'interacció

Per tal de simplificar el diagrama no s'ha inclòs el flux indicant que es pot anar tant a la Pantalla inicial com al registre d'usuaris des de qualsevol altra pantalla.



2.9 - Sketches

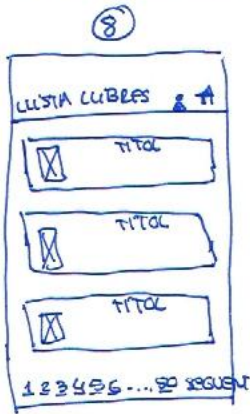




OBTINDRE INFO
D'UN LLIBRE OBTENENT
EL CODI QR DEL
LLOC DEL LLIBRE



TORNAR AL LLISTAT



AMPLIAR PI
RECORDAT
AMPLIAR PI
RECORDAT



2.10 - Prototipus d'alta fidelitat

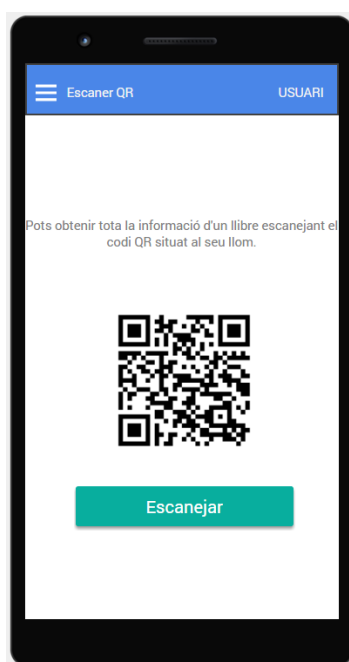
Comentaris del prototip:

- En la pantalla inicial, l'opció de llibres que tens en préstec únicament es mostraria si l'usuari està identificat a l'aplicació.
- En la pantalla de reserva de llibres, la informació sobre quan es pot recollir un llibre únicament apareix en cas que el llibre ja estigui reservat o en préstec i l'usuari s'incorpori a la cua de reserves.
- Tant a la imatge quatre com a la deu es mostren uns llistats de llibres en els quals es pot navegar a través del llistat pujant i baixant a la mateixa pantalla.
- Les imatges vuit i nou representen la mateixa pantalla, la vuit mostra si el llibre que s'ha reservat no estava prestat i la nou mostra si el llibre ja esta en préstec i s'ha afegit a l'usuari a la cua de reserves.
- Seguint les indicacions del consultor sobre disseny, es va modificar el prototipus inicial en els següents aspectes:
 - Eliminació del títol de l'App en totes les pantalles.
 - Substitució dels símbols Home (casa) i Usuari (persona), per un menú desplegable de tipus hamburguesa.
 - S'ha afegit informació d'utilitat per a l'usuari a la pantalla principal.

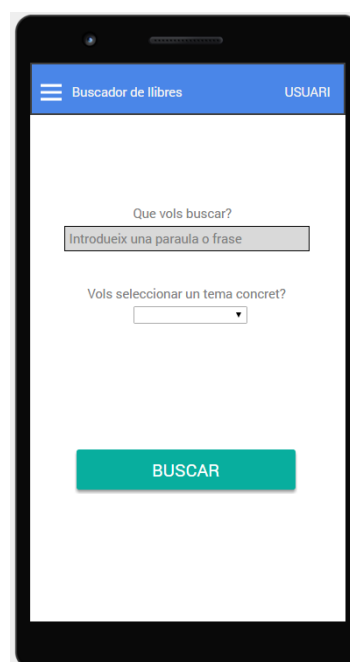
El prototipus final amb les millores indicades per el consultor es el següent:



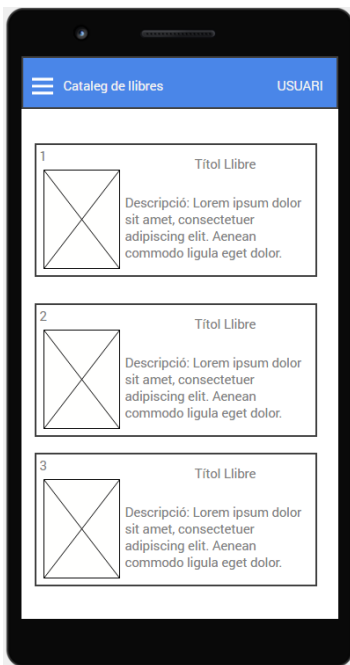
1 - Inici



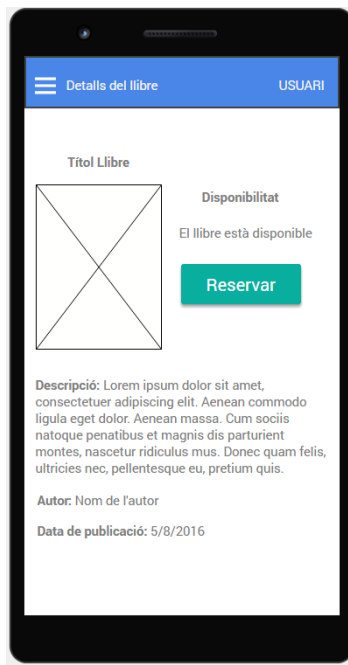
2- Escàner QR



3- Buscador de llibres



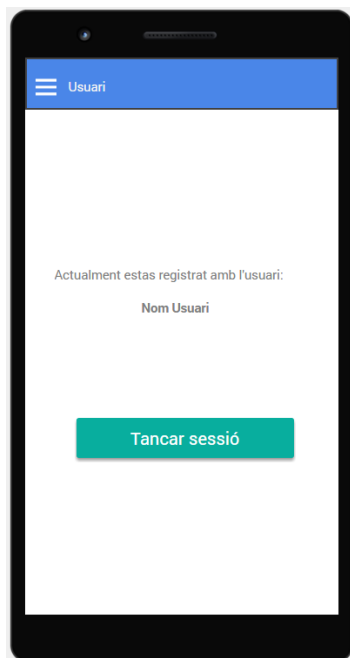
4- Catàleg de llibres



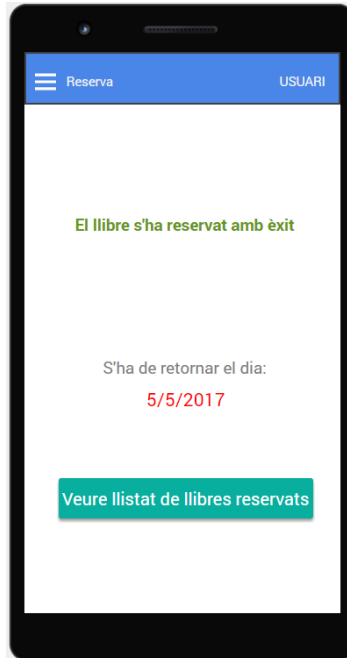
5- Detalls del llibre



6- Iniciar sessió



7- Tancar sessió



8- Reserva de llibres sense espera



9- Reserva de llibres amb espera



10- Llista de llibres reservats

2.11 - Avaluació

2.11.1 - Recull de preguntes:

1) Quina grau d'experiència te amb dispositius mòbils?

- Molt alt
- Alt
- Mitja
- Baix
- Molt baix

2) Te alguna dificultat a l'hora d'utilitzar el seu telèfon mòbil?

- Si
- No

3) Li agrada la lectura?

- Si
- No

4) Sol anar habitualment a alguna biblioteca?

- Si
- No

5) En cas afirmatiu sol utilitzar els serveis que ofereixen les biblioteques?

(Per exemple el préstec de llibres)

- Si
- No

6) Creu que li pot ser útil l'aplicació que s'està desenvolupant?

- Si
- No

2.11.2 - Tasques que haurien de realitzar els usuaris:

Les tasques que haurien de realitzar els usuaris es poden dividir en dos grans grups, per una banda les vuit primeres busquen saber si l'usuari entén la interfície gràfica i com interactuar amb ella, per altra banda en les tasques nou, deu i onze es vol saber si l'usuari sabria com navegar a través de l'aplicació.

Tasca 1: Observar la pantalla inicial

Pantalla inicial

Preguntes:

- Pot identificar quines són les funcionalitats principals que li ofereix l'aplicació en aquesta pantalla?
- Ha entès quines funcionalitats li proporcionen cada un dels botons?
- Ha vist el text USUARI la part superior dreta? Reconeix per a què serveix?

Tasca 2: Observar la pantalla de cerca de llibres

Pantalla cercador de llibres

Preguntes:

- Pot identificar quines són les funcionalitats principals que li ofereix l'aplicació en aquesta pantalla?
- Ha entès quines funcionalitats li proporcionen cada un dels elements de la pantalla? En cas negatiu, quin element no sap per a què serveix?
- Ha vist el símbol amb les tres barres horitzontals a la part superior esquerra? Reconeix per a què serveix?

Tasca 3: Observar la pantalla amb el llistat de llibres cercats.

Pantalla llista de llibres

Preguntes:

- Sabria com obtenir més informació d'un llibre?
- Troba a faltar alguna informació dels llibres que no apareix en aquesta pantalla?

Tasca 4: Observar la pantalla amb la informació d'un llibre en concret.

Pantalla detalls del llibre

Preguntes:

- Sabria com reservar el llibre que està veient?
- Sabria com tornar al llistat de llibres cercats?
- Troba a faltar alguna informació del llibre que està veient?

Tasca 5: Observar la pantalla per a iniciar sessió

Pantalla iniciar sessió

Preguntes:

- Sabria com iniciar sessió amb el seu usuari?

Tasca 6: Observar la pantalla per a tancar sessió

Pantalla tancar sessió

Preguntes:

- Sabria com tancar la sessió del seu usuari?

Tasca 7: Observar la pantalla de reserva de llibres

Pantalla reserva d'un llibre

Preguntes:

- Sabria quan ha de retornar el llibre que ha reservat?
- Entén perquè pot aparèixer un missatge indicant el dia a partir del qual es pot recollir el llibre?
- Sap que vol dir l'opció "Veure llistat de llibres reservats"?

Tasca 8: Observar la pantalla amb el llistat de llibres reservats que té l'usuari

Pantalla llistat de llibres reservats

Preguntes:

- Entén quina és la informació que s'està mostrant en aquesta pantalla?
- Troba a faltar alguna informació referent als llibres que té en préstec?
- Entén per a què serveix l'opció "Ampliar préstec"?
- Entén per a què serveix l'opció "Activar recordatori"?

- Entén per a què serveix l'opció "Veure detalls"?
- Sabria quan ha de retornar algun llibre que té en préstec?

Tasca 9: Observar la pantalla per a escanejar el codi QR d'un llibre

Pantalla escàner QR

Preguntes:

- Entén per a què serveix escanejar el codi QR d'un llibre?
- Sabria com escanejar el codi QR d'un llibre?
- Troba a faltar alguna altra indicació en aquesta pantalla referent al procés que ha de seguir per escanejar un codi QR?

Tasca 10: Partint de la pantalla inicial, intenti realitzar el procediment de cercar un llibre i reservar-lo.

Múltiples pantalles

Preguntes:

- Ha tingut alguna dificultat a l'hora de realitzar el procediment? En cas afirmatiu, quina dificultat ha tingut?
- Ha sentit que no sabia que estava fent en algun punt del procediment?
- Ha tingut alguna dificultat a l'hora d'entendre la informació que li estava proporcionant l'aplicació?
- En algun moment l'aplicació l'ha portat a alguna pantalla que no s'esperava?

Tasca 11: Partint de la pantalla inicial, sabria com identificar-se a l'aplicació amb el seu usuari i contrasenya i ampliar el temps de préstec d'un llibre?

Múltiples pantalles

Preguntes:

- Ha tingut alguna dificultat a l'hora de realitzar el procediment? En cas afirmatiu, quina dificultat ha tingut?
- Ha sabut com identificar-se a l'aplicació mitjançant el seu usuari i contrasenya?
- Ha trobat pogut trobar sense dificultat l'opció per ampliar el temps de préstec d'un llibre?

Tasca 12: Partint de la pantalla inicial, intenti obtenir la informació d'un llibre mitjançant el codi QR que té al lloc.

Múltiples pantalles

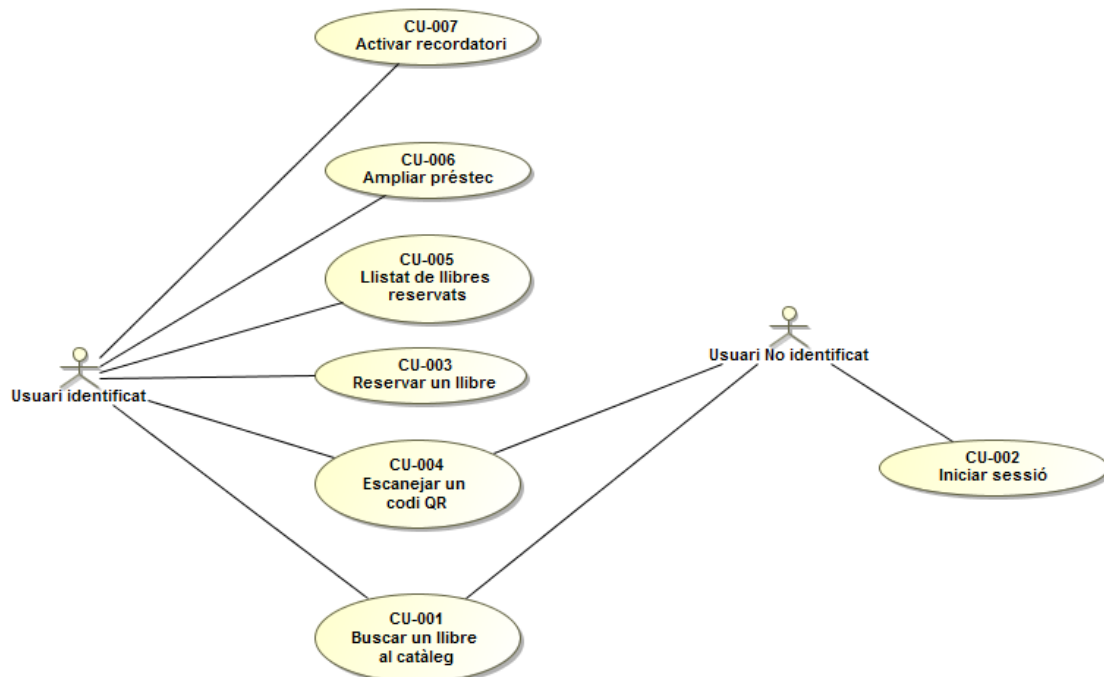
Preguntes:

- Ha tingut alguna dificultat a l'hora de realitzar el procediment? En cas afirmatiu, quina dificultat ha tingut?
- Ha entès les instruccions per a realitzar l'escaneig del codi QR?

3 - Disseny tècnic

3.1 - Diagrama de casos d'ús

A continuació es mostra el diagrama de casos d'ús on es distingeix els casos que pot dur a terme un usuari identificat i un usuari no identificat.



3.2 - Casos d'ús

Identificador	CU-001
Nom	Buscar un llibre al catàleg
Prioritat	Alta
Descripció	Es vol buscar la informació d'un llibre al catàleg de la biblioteca
Actors	Usuari identificat al sistema /Usuari no identificat al sistema
Pre-Condicions	Hi ha llibres al catàleg de la biblioteca
Iniciat per	Usuari identificat al sistema /Usuari no identificat al sistema
Flux	1) L'usuari selecciona l'opció "Buscador de llibres". 2) El sistema mostra la pantalla per a buscar llibres. 3) L'usuari introdueix una paraula clau i selecciona un tema. 4) El sistema mostra un llistat de llibres del tema seleccionat per l'usuari i amb el títol que contingui la paraula clau seleccionada.
Post-Condicions	-
Notes	-

Identificador	CU-002
Nom	Iniciar sessió
Prioritat	Alta
Descripció	Es vol iniciar sessió amb un usuari a l'aplicació
Actors	Usuari no identificat al sistema
Pre-Condicions	Hi ha un usuari creat al sistema.
Iniciat per	Usuari no identificat al sistema
Flux	1) L'usuari introdueix el seu nom d'usuari i contrasenya. 2) El sistema comprova que el nom d'usuari i la contrasenya concordin amb algun usuari registrat a l'aplicació. En cas que no concordi es torna al punt 1, en cas contrari es continua al punt 3. 3) El sistema inicia sessió amb l'usuari indicat i guarda la informació de l'usuari al telèfon.
Post-Condicions	La informació de l'usuari ha estat guardada al telèfon.
Notes	

Identificador	CU-003
Nom	Reservar un llibre
Prioritat	Alta
Descripció	Es vol reservar un llibre del catàleg de la biblioteca.
Actors	Usuari identificat al sistema /Usuari no identificat al sistema
Pre-Condicions	Hi ha llibres al catàleg de la biblioteca
Iniciat per	Usuari identificat al sistema /Usuari no identificat al sistema
Flux	1) L'usuari busca un llibre al catàleg (CU-001) i el selecciona per obtenir més informació. 2) El sistema mostra la informació del llibre seleccionat. 2.a) Si el llibre ja està en préstec el sistema ho indica a l'usuari. 3) L'usuari reserva el llibre. 3.a) Si l'usuari no ha iniciat sessió al sistema. 3.a1) El sistema indica a l'usuari que ha d'iniciar sessió.

	<p>3.a2) L'usuari inicia sessió al sistema (CU-002).</p> <p>3.a3) Es torna al punt 2.</p> <p>4) El sistema mostra la confirmació de la reserva juntament amb la data en què haurà de ser retornat el llibre.</p> <p>4.a) Si el llibre ja està en préstec el sistema inclou a l'usuari a la cua de reserves i indica a l'usuari a partir de quin dia podrà recollir el llibre i en quina data l'haurà de retornar.</p>
Post-Condicions	L'usuari identificat al sistema ha reservat un llibre del catàleg.
Notes	No es permet reservar un llibre a un usuari no identificat.

Identificador	CU-004
Nom	Escanejar un codi QR
Prioritat	Baixa
Descripció	Es vol obtenir la informació d'un llibre mitjançant l'escaneig del codi QR que té ubicat al seu lloc.
Actors	Usuari identificat al sistema /Usuari no identificat al sistema
Pre-Condicions	Hi ha llibres al catàleg de la biblioteca. Els llibres disposen d'un codi QR al seu lloc. Aquest codi QR enllaça amb la informació del llibre.
Iniciat per	Usuari identificat al sistema /Usuari no identificat al sistema
Flux	<p>1) L'usuari selecciona l'opció "Escanejar codi QR".</p> <p>2) El sistema mostra la pantalla amb les instruccions per escanejar el codi QR.</p> <p>3) L'usuari escaneja el codi QR.</p> <p>4) El sistema mostra la pantalla amb la informació del llibre identificat pel codi QR.</p>
Post-Condicions	-
Notes	-

Identificador	CU-005
Nom	Llistat de llibres reservats
Prioritat	Alta
Descripció	Es vol obtenir el llistat de llibres reservats que té un usuari.
Actors	Usuari identificat al sistema
Pre-Condicions	Hi ha llibres al catàleg de la biblioteca. Existeix un usuari registrat al sistema. L'usuari té, com a mínim, un llibre reservat.
Iniciat per	Usuari identificat al sistema
Flux	<p>1) L'usuari selecciona l'opció "Llibres que tens en préstec".</p> <p>2) El sistema mostra la pantalla amb el llistat de llibres en préstec de l'usuari.</p>
Post-Condicions	-
Notes	-

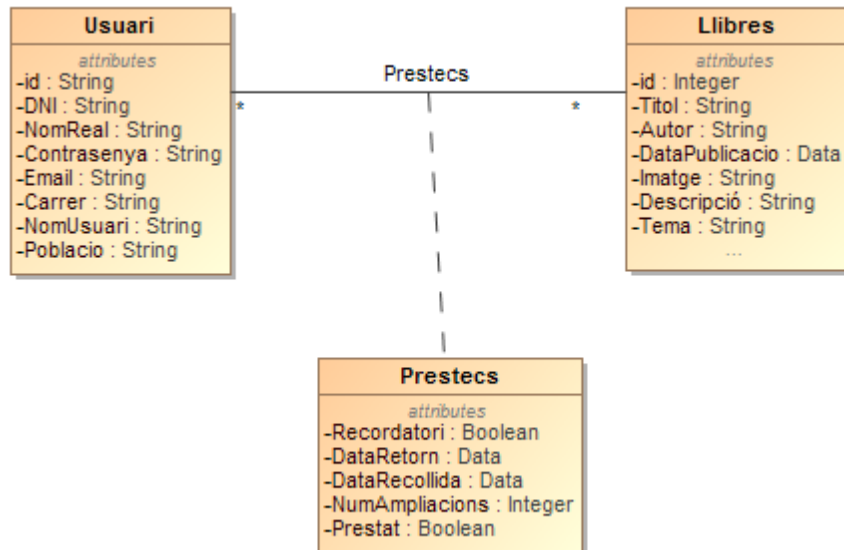
Identificador	CU-006
Nom	Ampliar préstec
Prioritat	Mitjana
Descripció	Es vol ampliar el temps de préstec d'un llibre reservat per l'usuari.
Actors	Usuari identificat al sistema
Pre-Condicions	Hi ha llibres al catàleg de la biblioteca. Existeix un usuari registrat al sistema. L'usuari té, com a mínim, un llibre reservat.
Iniciat per	Usuari identificat al sistema
Flux	1) L'usuari consulta el llistat de llibres reservats (CU-005). 2) L'usuari selecciona l'opció d'ampliar el temps de reserva d'un llibre del llistat. 3) El sistema amplia el temps de préstec del llibre, actualitza la data en què ha de ser retornat.
Post-Condicions	La data de retorn del llibre seleccionat s'ha ampliat.
Notes	Si el llibre té una altra reserva per part d'un altre usuari es deshabilita l'opció per ampliar el préstec. Per evitar un mal ús d'aquesta funcionalitat el llibre únicament podrà ser ampliat amb aquest procediment fins a tres vegades.

Identificador	CU-007
Nom	Activar recordatori
Prioritat	Mitjana
Descripció	Es vol activar una alarma per a que l'aplicació recordi a l'usuari el dia que ha de retornar un llibre que tingui en préstec.
Actors	Usuari identificat al sistema.
Pre-Condicions	Hi ha llibres al catàleg de la biblioteca. Existeix un usuari registrat al sistema. L'usuari té, com a mínim, un llibre reservat.
Iniciat per	Usuari identificat al sistema
Flux	1) L'usuari consulta el llistat de llibres reservats (CU-005). 2) L'usuari selecciona l'opció d'activar recordatori d'un llibre del llistat. 3) El sistema activa una notificació per al dia que ha de tornar el llibre l'usuari.
Post-Condicions	Es crea una notificació per al dia que l'usuari ha de retornar el llibre.
Notes	-

3.3 - Disseny de la base de dades

3.3.1 - Diagrama UML:

A continuació es mostra el diagrama UML de la base de dades:



3.3.2 - Model lògic relacional:

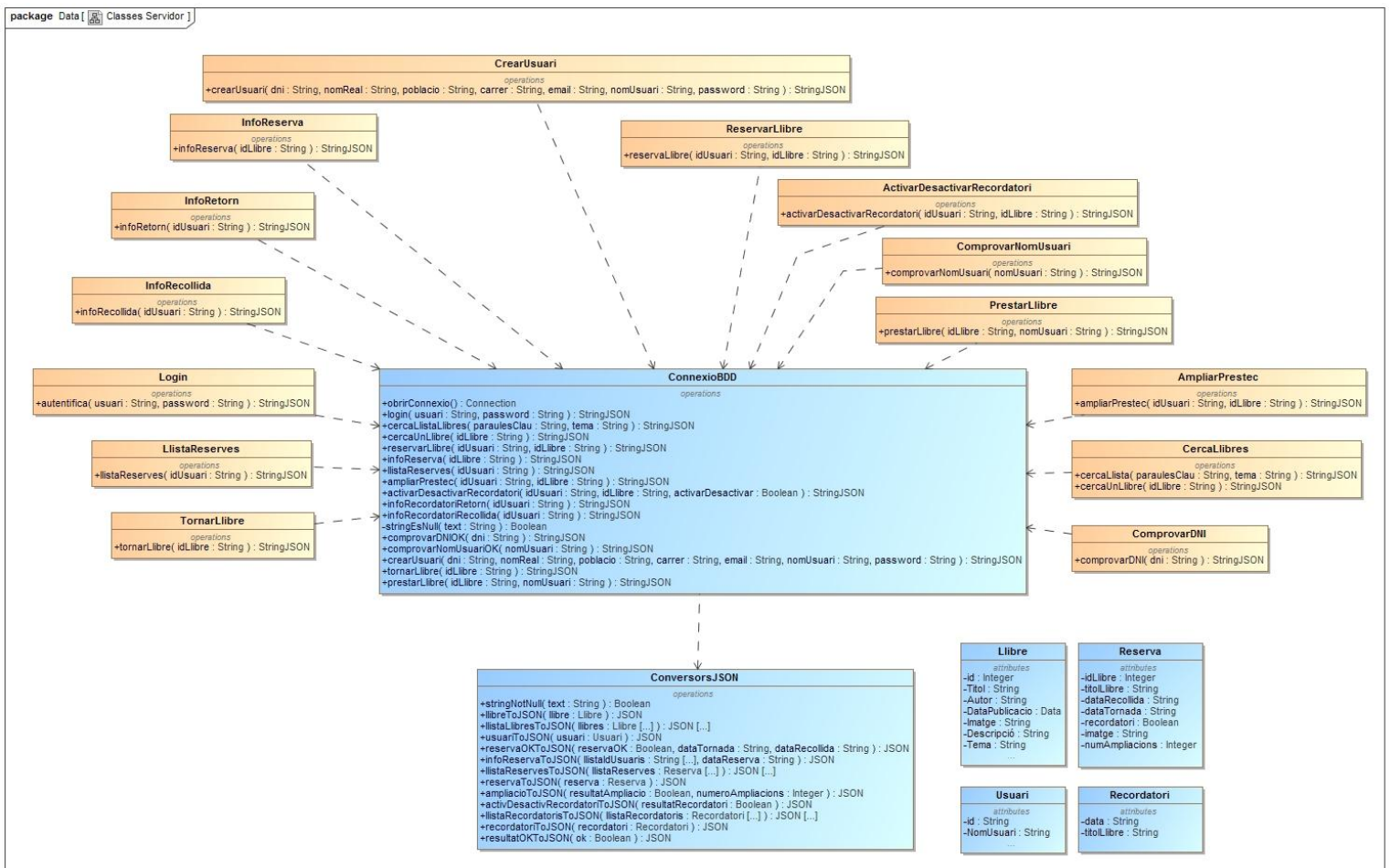
Usuari (**id : String**, DNI : String, NomReal: String, Contrasenya: String, Email: String, Carrer: String, NomUsuari : String, Poblacio: String)
DNI es un camp UNIC

Llibre (**id : Integer**, Títol: String, Autor: String, DataPublicació: Data, imatge: String, Descripció: String, Tema: String)

Prestecs (**IdUsuari: String, IdLlibre: Integer**, recordatori: Boolean, DataRetorn: Data, DataRecollida: Data, NumAmpliacions: Integer, Prestat: Boolean)
IdUsuari es clau foranea de Usuari.
IdLlibre es clau foranea de Llibre.

3.4 - Disseny de classes

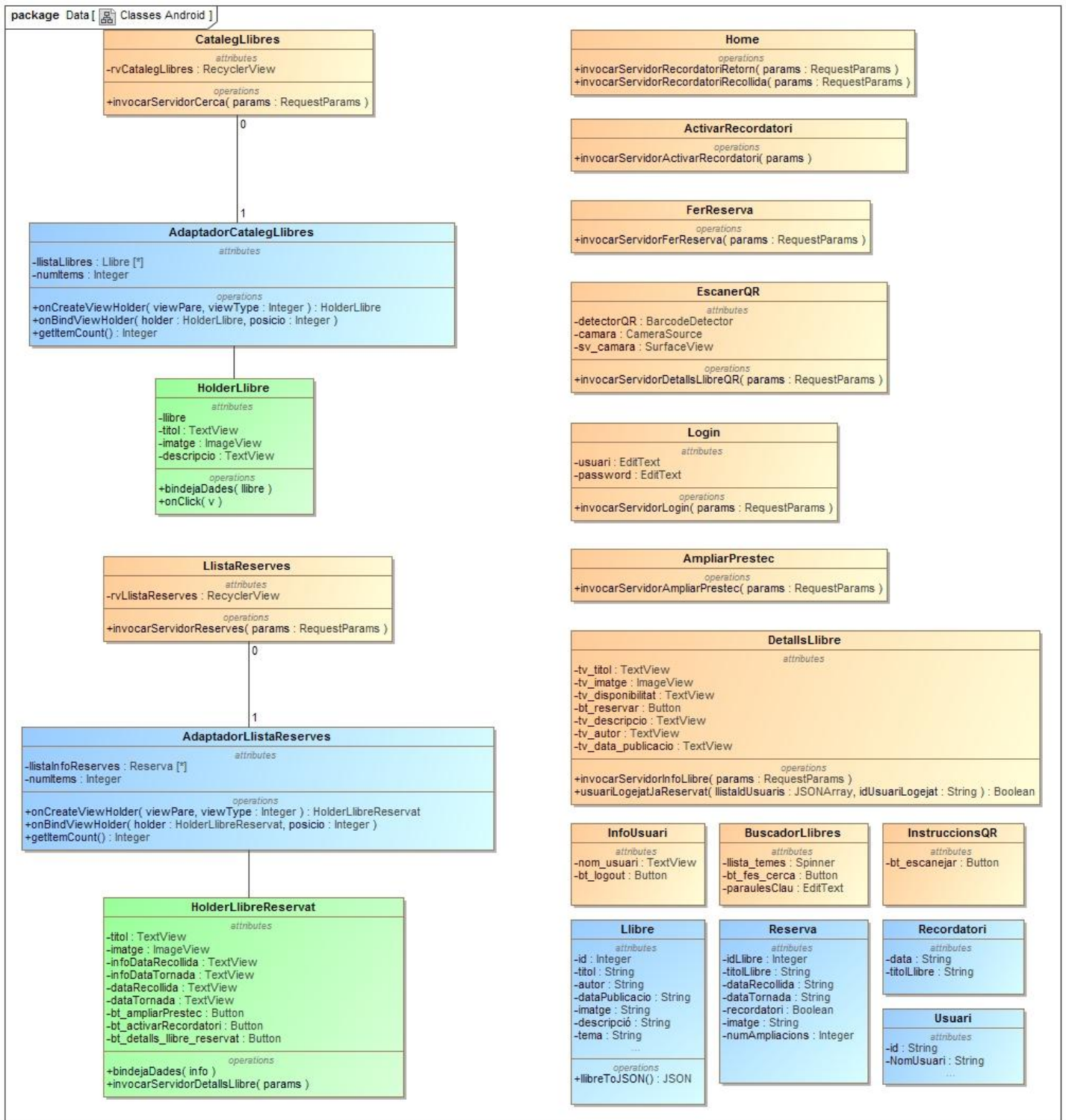
3.4.1 - Diagrama UML del servidor:



- Les classes de color taronja són les que representen els recursos publicats a Internet.
- Les classes de color blau són les encarregades de la gestió de dades i no es poden accedir des d'Internet directament.

3.4.2 - Diagrama UML de l'aplicació Android:

En aquest diagrama UML es mostren per una banda les activitats de l'aplicació amb els seus mètodes i atributs més destacats, per altra banda també es mostren les classes Java que no són activitats però estan encarregades d'organitzar la informació.



Les classes de color verd son classes Java internes (*nested class*) les quals estan definides al cos dels Adaptadors amb qui estan relacionades. Per altra banda les classes de color taronja son activitats i les de color blau son classes Java.

Cada activitat té un fitxer xml associat que determina com s'ha de mostrar en pantalla, no es mostren aquests elements de layout per tal de simplificar el diagrama i que sigui més entenedor.

S'ha omès els mètodes relacionats amb la generació i gestió dels menús tant superior com lateral a totes les activitats, ja que considero que no són rellevants per a descriure quin és el propòsit de l'activitat en si.

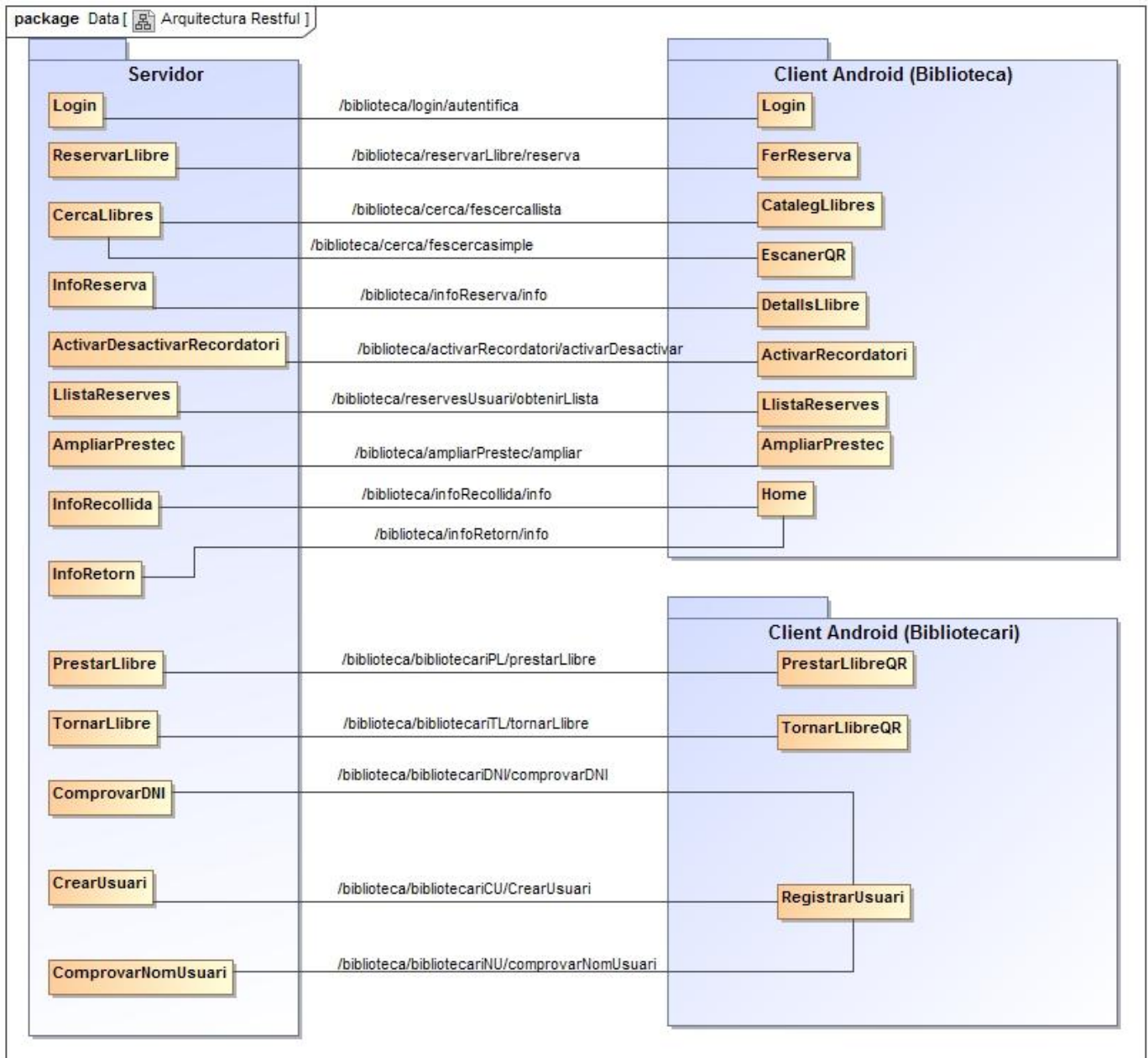
La majoria d'atributs de les classes que fan referència als elements de la interfície gràfica s'han omès, únicament estan inclosos els que he considerat més rellevants per entendre quin era la missió de la classe.

3.5 - Disseny de l'arquitectura

En aquest projecte s'ha utilitzat una arquitectura RESTful per a proporcionar connectivitat entre l'aplicació Android i el servidor amb la base de dades. Concretament l'arquitectura utilitzada te les següents característiques:

- S'utilitza HTTP per accedir als recursos publicats al servidor mitjançant el client Android a traves d'Internet.
- Aquests recursos son accessibles mitjançant una URI identificativa.
- S'accedeix als recursos mitjançant operacions GET.
- Les respostes del servidor estan codificades mitjançant cadenes de text en format JSON.

3.5.1 - Diagrama de l'arquitectura a alt nivell:



En aquest diagrama es mostra com les classes client invoquen mitjançant les URI indicades a sobre de cada relació les operacions del servidor.

Tot i que el client Bibliotecari està detallat en l'annex del projecte he decidit incloure'l en aquest diagrama perquè no es quedessin diverses classes del servidor sense proporcionar cap funcionalitat.

4 - Implementació

4.1 - Descripció d'eines, editors i API utilitzats

4.1.1 - Eines i editors

Per a la implementació del projecte s'ha utilitzat les següents eines i editors:

VirtualBox 4.2.18

S'ha decidit utilitzar una màquina virtual per crear el servidor de l'aplicació gràcies a la facilitat que hem presenta el poder anar fent còpies de seguretat de tot el servidor (sistema operatiu + configuració serveis + codi) utilitzant l'eina d'exportació de VirtualBox. A més en poder exportar la màquina virtual es pot transportar el servidor d'un ordinador a un altre sense problema ràpidament.

Es va escollir VirtualBox perquè és una eina amb la qual estic molt familiaritzar, per tant no es va considerar utilitzar alguna altra alternativa com VMware.

Eclipse Neon 4.6.3

S'ha utilitzat l'entorn IDE Eclipse per a desenvolupar tota la part de JavaEE del servidor. Igual que en el cas anterior es va escollir l'IDE Eclipse perquè és un entorn amb el qual estic molt familiaritzat, per tant tampoc es va considerar utilitzar alguna altra alternativa com NetBeans.

Android Studio 2.3

S'ha utilitzat l'entorn IDE Android Studio per a desenvolupar tota l'aplicació en Android. A més també s'ha utilitzat el simulador Android que proporciona l'IDE per a realitzar les proves a l'aplicació a mesura que s'anava desenvolupant.

Es va escollir Android Studio, ja que és l'eina més recomanada per desenvolupar una aplicació Android, incorpora moltes funcionalitats interessants com l'eina Gradle o la màquina virtual Android.

MySQL Community Server 5.7.18

S'ha utilitzat la base de dades MySQL per guardar tota la informació de la biblioteca, els seus usuaris i les reserves. Inicialment es va considerar l'opció d'utilitzar Firebase, el qual proporciona una base de dades de tipus NoSQL, no obstant a causa de les característiques de l'aplicació vaig decidir que era molt millor opció utilitzar una base de dades relacional com MySQL.

El principal motiu per descartar Firebase va ser que la base de dades que proporciona és molt poc flexible a l'hora de realitzar cerques per cadenes de text, el qual provocava que no fos possible implementar gaire correctament les cerques basades en els títols dels llibres.

Es va escollir MySQL principalment perquè igual que en els casos anteriors és un sistema gestor de bases de dades amb el qual ja estava molt familiaritzat.

Apache Tomcat 8.5.14

S'ha utilitzat el servidor Apache Tomcat per publicar el servei RestFul implementat al servidor. També és l'encarregat de servir les imatges dels llibres a l'aplicació, per tal de servir aquestes imatges es va haver de configurar un nou context en el fitxer server.xml indicant la ruta on estaven ubicades les imatges i la URL amb la qual es podrien accedir.

Es va escollir Apache Tomcat per la seva facilitat d'instal·lació i d'ús, a més al permetre proporcionar contingut estàtic juntament amb el servei RestFul el fa una eina ideal per al projecte.

Servei DDNS no - ip

S'ha utilitzat el servei no - ip per a proporcionar un DNS dinàmic al servidor, a causa de que la IP de casa meua és dinàmica s'utilitza aquest servidor DNS perquè l'aplicació pugui trobar el servidor a través d'Internet.

Es va escollir utilitzar aquest servei perquè ja hi havia treballat amb anterioritat i a més és gratuït i molt fiable.

4.1.2 - API utilitzades

Per a la implementació del projecte s'ha utilitzat les següents API:

Jersey 1.19.1

S'ha utilitzat Jersey per implementar el servidor RestFul amb Java EE, el qual proporciona a l'aplicació Android l'accés a la base de dades. Jersey és un framework open source que facilita molt la feina a l'hora d'implementar un servidor RestFul, ja que abstrau al programador de moltes de les complicacions que poden sorgir per comunicar client i servidor.

En aquest projecte en concret s'ha utilitzat les anotacions següents:

- *@Path*: Indica quina és la URI identificativa de la classe i dels seus mètodes amb la qual es pot accedir als mètodes definits a través de la xarxa.
- *@GET*: Indica que s'utilitza el mètode GET per accedir al recurs publicat a la xarxa.
- *@Produces*: Indica quin format tindrà la resposta del servidor al client, en el cas d'aquest projecte sempre es retornava en format JSON.
- *@QueryParam*: Indica en un mètode identificat per una URI un paràmetre que s'espera rebre en la petició GET del client. Si es vol que el mètode tingui diversos paràmetres s'han d'especificar tants *@QueryParam* com siguin necessaris.

Connector JDBC Mysql 5.1.17

S'ha utilitzat el driver proporcionat per MySQL per accedir a la base de dades mitjançant Java. L'accés a la base de dades està centralitzat en la classe ConnexioBDD.java del servidor, la qual s'encarrega d'executar totes les consultes mitjançant els mètodes següents:

- `executeQuery`: En el cas d'executar una consulta de tipus SELECT
- `executeUpdate`: En el cas d'executar consultes de tipus UPDATE, DELETE o INSERT.

Android Asynchronous Http Client 1.4.9

S'ha utilitzat la llibreria `android-async-http` per a realitzar les consultes i capturar les respostes del servidor des d'Android.

Aquesta llibreria proporciona un sistema molt senzill d'utilitzar, es basa en crear un client HTTP asíncron juntament amb un Handler en el qual es sobreescrueixen (Override) els mètodes `onSuccess` i `onFailure`, d'aquesta forma si la petició al servidor té èxit (es retorna el codi HTTP 200) s'executa el mètode `onSuccess` i en cas contrari s'executa el mètode `onFailure`.

A més en utilitzar aquesta llibreria no he hagut de preocupar-me per realitzar les invocacions HTTP en un thread diferent del de la interfície gràfica, ja que tots aquests aspectes estan ja implementats en la pròpia llibreria.

Glide 3.7.0

S'ha utilitzat la llibreria `Glide` per tal d'obtenir les imatges dels llibres amb Android des del servidor Tomcat.

En aquest cas aquesta llibreria també ha sigut molt útil i fàcil d'utilitzar, mitjançant els mètodes `load` i `placeholder` es pot obtenir una imatge publicada al servidor Tomcat i en cas que la imatge no estigui disponible o es carregui lentament es mostra una imatge substituïda per defecte. A més incorpora funcionalitats per guardar en cache imatges ja carregades, el qual és molt útil per reduir la càrrega del servidor.

Play-services-vision 9.6.1

S'ha utilitzat la llibreria `Vision`, la qual és una API proporcionada per Google, per implementar el lector de codis QR.

Aquesta llibreria proporciona accés a la càmera juntament amb un detector de codis QR mitjançant les classes `CameraSource` i `BarcodeDetector`, és una mica més complicada d'utilitzar que les API descrites anteriorment, ja que s'ha de sobreescrueixre un conjunt relativament gran de mètodes. A més abans de poder accedir a l'activitat lectora de codis s'ha de preguntar a l'usuari si vol donar permís a l'aplicació per a que pugui realitzar fotos i gravar vídeos.

4.2 - Decisions preses

4.2.1 - Implementació d'un servidor propi

En aquest projecte s'ha implementat un servidor propi utilitzant Tomcat, Jersey i MySQL. He pensat que aquesta era l'opció que tenia més sentit, ja que si l'aplicació ha de ser per a gestionar els llibres d'una biblioteca en concret, crec que el normal és tenir una base de dades amb els llibres que té la biblioteca i no tenia molt sentit per exemple obtenir llibres d'alguna de les API d'Internet com Google Books. Ara bé, en ser una biblioteca fictícia no es té llibres amb els quals omplir la base de dades, per tant he agafat una mostra de llibres d'Internet (Google Books) i he omplert la base de dades amb la seva informació com a mostra.

Tal com s'ha comentat anteriorment en primer lloc per a implementar el servidor vaig provar l'opció de Firebase però en tenir una base de dades NoSQL, la qual no permet fer cerques textuais gaire potents, vaig decidir descartar aquesta opció, ja que per exemple no podia implementar gaire correctament el buscador de llibres basant la cerca en el títol del llibre.

En descartar Firebase vaig decidir que la millor opció tal com ja he dit era implementar el meu propi servidor, ja que d'aquesta forma podria tenir el control complet de com és la base de dades i de quins serveis es proporcionen.

4.2.2 - Utilització de les API

He decidit utilitzar les API mencionades en l'apartat anterior ja que facilitaven molt la feina que havia de realitzar, per exemple gràcies a android-async-http he pogut implementar les peticions al servidor RestFul molt fàcilment sense haver de preocupar-me d'invocar manualment la petició Http en un thread diferent del que mostra la interfície gràfica i després capturar i retornar la informació.

4.2.3 - Creació d'usuaris

Inicialment vaig decidir no implementar un sistema de creació d'usuaris a l'aplicació, únicament es donava l'opció de fer login amb un usuari ja existent a la base de dades. Vaig decidir així perquè crec que té més sentit que un usuari abans de poder reservar un llibre hagi d'anar personalment a la biblioteca i que li creïn l'usuari allí físicament, agafant les seves dades com DNI i domicili, ja que en estar (suposadament) prestant llibres físics és més segur tenir les dades reals dels usuaris a qui els hi prestes.

Segons el meu punt de vista no té molt sentit que un usuari pugui registrar-se a l'aplicació autònomament i pugui començar a reservar llibres, ja que no hi hauria forma de saber si aquestes reserves són legítimes o bé si es tracta d'un usuari malintencionat el qual no té cap intenció d'anar a buscar els llibres que reserva.

Per altra banda també considero que tampoc té molt sentit oferir una opció per a donar-se d'alta a l'aplicació però deshabilitant les opcions que pot dur a terme un usuari registrat fins que l'usuari no valides el seu compte físicament a la biblioteca, ja que amb quin motiu voldria l'usuari registrar-se si després no pot dur a terme cap funcionalitat fins que no anés físicament a la biblioteca.

No obstant, en una valoració del projecte el consultor em va recomanar que afegís aquesta funcionalitat al projecte, aleshores el que he fet ha sigut desenvolupar una petita aplicació per al bibliotecari de la biblioteca on implemento aquesta funcionalitat, d'aquesta forma segueixo amb la meua idea sobre que és el millor per a l'aplicació tot i incorporant la funcionalitat al projecte desenvolupat.

Tant a l'apartat 4.2.6 com a l'annex del projecte s'explica amb més detall el raonament per haver desenvolupat aquesta aplicació bibliotecari juntament amb la informació més rellevant d'aquesta aplicació.

4.2.4 - Ampliació de préstec de llibres

He decidit que un usuari no pugui ampliar més de 3 cops el termini de préstec d'un llibre que té reservat, ja que sinó podria abusar del sistema i tenir el llibre reservat per a ell massa temps. A més tampoc es deixa ampliar el termini de préstec si hi ha un altre usuari esperant a que es retorni el llibre.

4.2.5 - Categories de cerca

Únicament s'ha inclòs dues categories de llibres a l'aplicació (Informàtica i Matemàtiques), amb deu llibres a cada categoria, considero que és un catàleg suficient per a poder provar les funcionalitats de l'aplicació però no descarto ampliar-lo en cas que sigui necessari.

4.2.6 - Aplicació bibliotecari

Un cop ja tenia pràcticament acabada l'aplicació principal del projecte vaig tenir la sensació que l'aplicació estava poc arrodonida i que presentaria un projecte com "a mitges", ja que tot i proporcionar les funcionalitats descrites en els apartats anteriors de la memòria, realment no tenia cap funcionalitat amb la qual prestar un llibre a un usuari o tornar un llibre prestat a la biblioteca, a més tampoc estava incorporant l'alta d'usuaris.

Per tant vaig pensar que el que tenia més sentit era crear una petita aplicació com annex al projecte per al bibliotecari de la biblioteca, amb aquesta aplicació el bibliotecari pot prestar un llibre a un usuari, tornar un llibre a la biblioteca i donar d'alta usuaris, d'aquesta forma podia seguir per una banda amb la idea exposada al punt 4.2.3 i per altra banda podia incorporar informació més valuosa a la pantalla inicial (Home) tal com suggeria el consultor de disseny en una de les valoracions del projecte.

En l'annex del projecte es pot trobar informació més detallada sobre aquesta aplicació bibliotecari.

4.2.7 - Retorn de llibres

En el moment que un usuari1 retorna un llibre, si hi ha un altre usuari2 esperant que el llibre sigui retornat se'l notifica i es canvia la data de recollida del llibre per a l'usuari2 a la data actual.

He decidit canviar únicament la data de recollida i no la de tornada del usuari2, ja que si un usuari té previst tenir un llibre fins a un dia en concret, no és gaire just reduir aquests dies únicament perquè l'usuari anterior hagi retornat el llibre abans de temps.

4.2.8 - Home sense menú

He decidit que la pàgina inicial de l'aplicació no tingui implementat el menú de navegació que si tenen totes les altres pàgines, considero que no te gaire sentit tenir el menú de navegació en aquesta activitat degut a que ja es pot accedir a totes les funcionalitats principals directament amb els botons que s'ofereixen.

4.3- Codi font Android

A continuació es detallarà els aspectes principals del codi font de l'aplicació Android, únicament s'inclouen fragments de codi que es consideren especialment rellevants.

4.3.1 - RecyclerView i Adaptadors

El RecyclerView és un widget proporcionat per l'API d'Android que permet mostrar llistes molt grans de dades de forma eficient. Es va decidir utilitzar aquest widget tant per mostrar el catàleg de llibres de la biblioteca (CatalogLlibre) com per a mostrar la llista de llibres que un usuari te reservats (ListaReserves), ja que en un entorn real és molt possible que aquestes llistes poguessin arribar a ser molt grans, especialment la del catàleg de llibres.

A més proporcionen una interacció amb l'usuari bastant còmoda, ja que es pot anar desplaçant amunt i avall de la llista sense haver d'esperar a cap recarrega ni a canviar de pantalla.

Per tal d'utilitzar aquest widget s'ha de definir una classe adaptador i un administrador de disseny (LayoutManager). En el cas d'aquest projecte les classes adaptador son AdaptadorCatalogLlibres per al RecyclerView de l'activitat CatalogLlibre i AdaptadorCatalog Reserves per al RecyclerView de l'activitat LlistaReserves.

A continuació es detallarà com exemple el RecyclerView encarregat del catàleg de llibres, la classe AdaptadorCatalogLlibres es l'encarregada d'indicar com s'ha de mostrar el llistat de llibres que se li passa per paràmetre al RecyclerView, per tal d'implementar l'adaptador s'ha d'especificar el següent:

Atributs i constructor:

```
public class AdaptadorCatalogLlibres extends
RecyclerView.Adapter<AdaptadorCatalogLlibres.HolderLlibre> {

    //Contindrà la llista de llibres que s'ha obtingut del servidor
    private ArrayList<Llibre> llistaLlibres = new ArrayList<Llibre>();

    //Contindrà la grandària de la llista de llibres obtinguda del servidor
    int numItems;

    /*
    *Constructor que rep com a paràmetre la llista de llibres que es mostraran
    */
    public AdaptadorCatalogLlibres(ArrayList<Llibre> llistaLlibres){
        this.llistaLlibres = llistaLlibres;
        this.numItems = llistaLlibres.size();
    }
}
```

S'ha de sobreescrivre els mètodes onCreateViewHolder, onBindViewHolder i getItemCount

```
/*
 * Mètode que es crida quan es crea un nou ViewHolder el qual contindrà la
 * vista per a cada ítem de la llista.
 * Te com a paràmetre el ViewGroup en el qual està contingut i un int amb
 * el que es podria canviar el layout.
 */

@Override
public HolderLlibre onCreateViewHolder(ViewGroup viewPare, int viewType){
    LayoutInflator inflador = LayoutInflater.from(viewPare.getContext());
    View view = inflador.inflate(R.layout.item_catalegllibres, viewPare, false);
    HolderLlibre holderLlibre = new HolderLlibre(view);
    return holderLlibre;
}

/*
 * Mètode el qual crida el RecyclerView per a mostrar les dades dels ítems
 * en una posició determinada.
 * Te com a paràmetres el viewHolder que ha de ser actualitzat i la posició
 * que ocupa en la llista el ítem corresponent
 */
@Override
public void onBindViewHolder(HolderLlibre holder, int posicio){
    Llibre llibreActual = llistaLlibres.get(posicio);
    holder.bindejaDades(llibreActual);
}

@Override
public int getItemCount() {
    return numItems;
}
```

S'ha de crear una classe interna ViewHolder que definirà com es mostra cada element de la llista, a més s'encarrega de lligar les dades de l'element de la llista amb la interfície gràfica.

```
/*
 * Classe interna que defineix el ViewHolder utilitzat per aquest adaptador
 * de RecyclerView
 *
 * */
class HolderLlibre extends RecyclerView.ViewHolder implements
View.OnClickListener
{
    //Aquesta variable llibre serà utilitzada per a ser passada com a paràmetre a
    la nova activitat encarregada de mostrar els detalls del llibre seleccionat

    Llibre llibre;
    TextView titol;
    ImageView imatge;
    TextView descripcio;

    /*
     * Constructor per al ViewHolder de llibres, definim un onClickListener
     * per definir el comportament dels clicks a la llista en el mètode
     * onClick
     * */
    public HolderLlibre(View itemView){
        super(itemView);
        titol = (TextView) itemView.findViewById(R.id.rv_item_titol);
        imatge = (ImageView) itemView.findViewById(R.id.rv_item_imatge);
        descripcio = (TextView) itemView.findViewById(R.id.rv_item_descripcio);
        itemView.setOnClickListener(this);
    }
}
```

```

    /* Mètode que col·loca les dades del del llibre de la posició actual a
    dintre de les diferents vistes del viewHolder*/
    private void bindejaDades(Llibre llibre){
        this.llibre=llibre;
        titol.setText(llibre.getTitol());
        descripcio.setText(llibre.getDescripcio());

    }

    // Carreguem la imatge fent servir la llibreria Glide, en cas que la imatge no
    estigui disponible es mostra una imatge d'error
    Context contextImatge = imatge.getContext();
    Glide.with(contextImatge).load(contextImatge.getString(R.string.domini_servido
    r)+"/imatges/" + llibre.getImatge())
        .placeholder(R.drawable.img_no_disponible)
        .error(R.drawable.img_no_disponible)
        .into(imatge);
    }

    /**Mètode que defineix el comportament del Click en un ítem.
    * En aquest cas s'obre una activitat nova passant com a paràmetre el
    llibre seleccionat en format JSON
    */
    @Override
    public void onClick(View v) {
        Context context = itemView.getContext();
        Intent nouIntent = new Intent(context, DetallsLlibre.class);
        nouIntent.putExtra("Llibre", llibre.llibreToJson().toString());
        context.startActivity(nouIntent);
    }
}
}
}

```

Un cop implementat l'adaptador junt amb el seu ViewHolder, ja s'està preparat per utilitzar la RecyclerView, únicament falta per una banda vincular l'adaptador amb el layout del RecyclerView a la classe CatalegLlibres.

```

adaptadorCatalegLlibres = new AdaptadorCatalegLlibres(llistaLlibres);
rvCatalegLlibres.setAdapter(adaptadorCatalegLlibres);

```

I per altra banda com a pas final es defineix quin és el LayoutManager que es vol fer servir, en aquest cas s'ha utilitzat el LinearLayoutManager, el qual és un administrador de disseny que ja ve incorporat amb el propi widget. S'utilitza aquest administrador per mostra els elements de la llista en format vertical.

```

rvCatalegLlibres.setLayoutManager(newLinearLayoutManager(CatalegLlibres.this))

```


4.3.2 - Detalls d'un llibre i invocació del servidor

Tal i com s'ha mencionat anteriorment per tal de realitzar les invocacions al servidor s'ha utilitzat la llibreria Android Asynchronous Http Client 1.4.9, a continuació es detalla el codi que s'utilitza a la classe DetallsLlibre per obtenir les dades d'un llibre en concret des del servidor. També es mostra la lògica que es segueix per determinar quin missatge s'ha de mostrar a l'usuari referent a l'estat actual de disponibilitat del llibre.

```
/*La petició per obtenir la informació de la reserva del llibre s'executa onStart() per tal de que
s'actualitzi al tornar a aquesta activitat si un usuari logeja a l'aplicació*/
//Executem la creació del menu onStart per si un usuari logeja al telefon actualitzar el menu i
mostrar la opció d'anar al llistat de llibres en presteo
@Override
protected void onStart() {
    super.onStart();
    afegirMenuNavegacio(); //Metode que configura el menu de navegació
    RequestParams params = new RequestParams();
    params.put("idLlibre", objLlibre.getId());
    invocarServidorInfoLlibre(params);
}

/*Aquest metode s'encarrega d'obtenir la informacio de reserva del llibre i actualitzar la
interficie en cas necessari*/
private void invocarServidorInfoLlibre(RequestParams params){

    dialegProgres.show();

    AsyncHttpClient client = new AsyncHttpClient();
    client.get(getString(R.string.domini_servidor)+"/biblioteca/infoReserva/info",params ,new
AsyncHttpResponseHandler()
{

/*S'executa el metode onSuccess si la resposta rebuda conte el codi de resposta Http 200
*/
@Override
public void onSuccess(int statusCode, Header[] headers, byte[] response) {
    dialegProgres.hide();
    tv_disponibilitat.setTextColor(Color.BLACK);
    bt_reservar.setVisibility(View.VISIBLE);

    try
    {
        //Obtenim la resposta del servidor
        JSONObject infoLlibre = new JSONObject(new String(response, "UTF-8"));

        //La resposta conte la ultima data de la cua de reserves
        String dataReserva = infoLlibre.getString("dataReserva");
        paramDataReserva = dataReserva;

        //Juntament amb els id de tots els usuaris que tenen el llibre reservat
        JSONArray llistaIdUsuaris = infoLlibre.getJSONArray("llistaIdUsuaris");

        //Obtenim el id del usuari logejat al telefon
        SharedPreferences sp= getSharedPreferences("Login",0);
        String idUsuariLogejat = sp.getString("idUsuari",null);
    }
}
}
}
```

```

//Si NO hi ha cap usuari logejat al telefon ->
if(idUsuariLogejat == null || idUsuariLogejat.equals("-1")) {
    //i el llibre no te cap reserva ->
    if (dataReserva.equals("") || dataReserva == null){
        tv_disponibilitat.setText("El llibre esta disponible");
    }
    //En canvi si el llibre te alguna reserva mostrem la data en la que estara disponible
    else{
        tv_disponibilitat.setText("El llibre no esta disponible fins al: \n"+dataReserva);
        tv_disponibilitat.setTextColor(Color.RED);
    }
}
//Si hi ha un usuari logejat al telefon ->
else{
    //Pero el llibre no te cap reserva ->
    if (dataReserva.equals("") || dataReserva == null){
        tv_disponibilitat.setText("El llibre esta disponible");
    }

    //Si el llibre si te alguna reserva, comprovem si una d'elles es la del propi usuari
    logejat
    else if(usuariLogejatJaReservat(llistaIdUsuaris, idUsuariLogejat)) {
        tv_disponibilitat.setText("Ja tens reservat aquest llibre");
        bt_reservar.setVisibility(View.INVISIBLE);
    }
    //En cas que l'usuari logejat no tingui cap reserva d'aquest llibre ->
    else{
        tv_disponibilitat.setText("El llibre no esta disponible fins al: \n"+dataReserva);
        tv_disponibilitat.setTextColor(Color.RED);
    }
}

} catch (Exception e){
    Log.d("Error", "Error en la resposta del servidor ");
    e.printStackTrace();
}

}

/*Si la resposta del servidor NO conte el codi Http de resposta 200 s'executa el metode
onFailure*/
@Override
public void onFailure(int statusCode, Header[] headers, byte[] errorResponse, Throwable e)
{
    dialogProgres.hide();
    tv_disponibilitat.append("No es pot obtenir les dades de reserva");
    bt_reservar.setVisibility(View.INVISIBLE);
    //Dependent del codi de resposta d'error indiquem un missatge o un altre

    if(statusCode == 500){
        Toast.makeText(getApplicationContext(), "500. Error del servidor",
        Toast.LENGTH_LONG).show();
    }
    else if(statusCode == 404){
        Toast.makeText(getApplicationContext(), "404. Recurs no trobat",
        Toast.LENGTH_LONG).show();
    }
    else{
        Toast.makeText(getApplicationContext(), "Dispositiu sense connexió o servidor no
        disponible", Toast.LENGTH_LONG).show();
    }
}
});
}
}

```

4.3.3 - Login / Logout i emmagatzemament de l'usuari al telèfon

A continuació es mostra el codi encarregat tant d'iniciar sessió (Login) com de tancar sessió (Logout). També es mostra com s'emmagatzema l'usuari al telèfon un cop s'inicia sessió.

L'activitat encarregada de realitzar l'inici de sessió s'anomena Login, únicament es mostren les parts del codi més rellevants:

```
@Override
protected void onCreate(Bundle savedInstanceState) {

    (...)

    //Definim el comportament del boto de login
    bt_login.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View v) {
            /*En cas que el servidor no sigui instantani retornant les dades es mostra un missatge
            a l'usuari indicant que s'esta treballant en la seva petició*/
            dialegProgres = new ProgressDialog(Login.this);
            dialegProgres.setMessage("Autenticant usuari...");
            dialegProgres.setCancelable(false);

            /*Realitzem la petició de login al servidor amb els parametres indicats*/
            RequestParams params = new RequestParams();
            params.put("usuari", usuari.getText());
            params.put("password", password.getText());
            invocarServidorLogin(params);
        }
    });
    (...)
}

private void invocarServidorLogin(RequestParams params){

    dialegProgres.show();

    AsyncHttpClient client = new AsyncHttpClient();
    client.get(getString(R.string.domini_servidor)+"/biblioteca/login/autentifica",params ,new
    AsyncHttpResponseHandler()
    {

        /*S'executa el metode onSuccess si la resposta rebuda conte el codi de resposta Http 200*/
        @Override
        public void onSuccess(int statusCode, Header[] headers, byte[] response) {

            dialegProgres.hide();
            Usuari usuari;
            try {

                //Obtenim la resposta del servidor
                JSONObject usuariJson = new JSONObject(new String(response, "UTF-8"));
                usuari = new Usuari(usuariJson.getString("id"), usuariJson.getString("nomUsuari"));

                //Si les dades proporcionades no corresponen a cap usuari idUsuari == -1
                if(usuari.getIdUsuari().equals("-1")){

                    Toast.makeText(getApplicationContext(), "Error. Les dades proporcionades no
                    corresponen a cap usuari", Toast.LENGTH_LONG).show();
                }
            }
        }
    }
}
```

```

//En cas contrari guardem el usuari logejat al telefon
else
{
    SharedPreferences sp = getSharedPreferences("Login", 0);
    SharedPreferences.Editor Ed = sp.edit();
    Ed.putString("idUsuari",usuari.getIdUsuari());
    Ed.putString("nomUsuari",usuari.getNomUsuari());
    Ed.commit();

    Toast.makeText(getApplicationContext(), "Sessió oberta correctament",
    Toast.LENGTH_LONG).show();
    finish();
}

}catch (Exception e){
    Log.d("Error","Error en la resposta del servidor");
    e.printStackTrace();
}
}

/*Si la resposta del servidor NO conte el codi Http de resposta 200 s'executa el metode
onFailure*/
@Override
public void onFailure(int statusCode, Header[] headers, byte[] errorResponse, Throwable e)
{
    dialegProgres.hide();
    //Depenent del codi de resposta d'error indiquem un missatge o un altre

    if(statusCode == 500){
        Toast.makeText(getApplicationContext(), "500. Error del servidor",
        Toast.LENGTH_LONG).show();
    }
    else if(statusCode == 404){
        Toast.makeText(getApplicationContext(), "404. Recurs no trobat",
        Toast.LENGTH_LONG).show();
    }
    else{
        Toast.makeText(getApplicationContext(), "Dispositiu sense connexió o servidor no
        disponible", Toast.LENGTH_LONG).show();
    }
}
});
}
}

```

Com es pot veure s'utilitza les SharedPreferences per guardar el id d'usuari i el nom d'usuari al telèfon, d'aquesta forma encara que es tanqui l'aplicació o fins i tot es tanqui el telèfon l'usuari es queda emmagatzemat a un fitxer local del telèfon i pot ser recuperat sense necessitat que l'usuari hagi d'estar iniciant sessió cada cop que obre l'aplicació.

Per a tancar sessió a l'aplicació s'utilitza l'activitat InfoUsuari, la qual té un botó que elimina a l'usuari emmagatzemat de les SharedPreferences.

```

public class InfoUsuari extends AppCompatActivity {

    (...)

    bt_logout.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View v) {

            //Eliminem l'usuari de les sharedPreferences, per tant fem logout amb ell.
            SharedPreferences preferences = getSharedPreferences("Login",0);
            SharedPreferences.Editor editor = preferences.edit();
            editor.clear();
            editor.commit();

            Toast.makeText(getApplicationContext(), "Sessió tancada correctament",
            Toast.LENGTH_LONG).show();
            finish();
        }
    }
}

```

4.3.4 - Sol·licitar permisos per accedir a la càmera

Per tal de poder utilitzar el lector de codis QR l'aplicació ha de sol·licitar permís a l'usuari per poder accedir a la càmera del dispositiu. A continuació es mostra el codi necessari per a realitzar aquesta sol·licitud:

En el fitxer AndroidManifest.xml s'ha d'incloure el següent:

```
(...)  
  
<uses-feature  
    android:name="android.hardware.camera"  
    android:required="true" />  
  
<uses-permission android:name="android.permission.INTERNET" />  
<uses-permission android:name="android.permission.CAMERA" />  
  
(...)
```

També s'ha de comprovar abans d'accedir a l'activitat que requereix l'accés a la càmera si ja es tenen els permisos, en cas que no es tinguin es sol·liciten, en canvi si ja es tenen s'accedeix a l'activitat lectora de codis QR directament.

```
//Comprovem que tinguem permisos per accedir a la camera, si no els tenim  
els demanem  
if (ContextCompat.checkSelfPermission(InfoUsuari.this,  
Manifest.permission.CAMERA) != PackageManager.PERMISSION_GRANTED)  
{  
  
    ActivityCompat.requestPermissions(InfoUsuari.this,  
                                     new String[]{Manifest.permission.CAMERA},  
                                     1);  
}  
//Si ja tenim permisos anem directament al lector QR  
else{  
    startActivity(new Intent(InfoUsuari.this, InstruccionsQR.class));  
}
```

Si l'usuari encara no ha proporcionat els permisos a l'aplicació, un cop decideix si els vol proporcionar o no s'executa el següent mètode:

```
//Metode que s'executa quan l'usuari respon a la otorgació o denegació de permisos  
@Override  
public void onRequestPermissionsResult(int requestCode, String permissions[], int[] grantResults)  
{  
    switch (requestCode) {  
        case 1: {  
            //Si l'usuari ha donat permisos obrim el lector QR  
            if (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION_GRANTED)  
{  
  
                startActivity(new Intent(InfoUsuari.this, InstruccionsQR.class));  
            } else {  
                //En cas contrari mostrem un missatge informant que la opció no esta habilitada  
                Toast.makeText(getApplicationContext(), "S'ha denegat el permis per accedir a la  
camera", Toast.LENGTH_LONG).show();  
            }  
        }  
    }  
}
```

Aquesta comprovació de permisos es realitza tant al botó per accedir al lector QR de l'activitat Home, com a totes les activitats que implementen el menú de navegació lateral i disposen de l'opció per viatjar directament al lector de codis QR

4.4- Codi font del Servidor Restful

En el codi font del servidor es pot distingir clarament dos tipus de classes, per una banda hi ha les classes públiques a través de la xarxa, encarregades de proporcionar cada una d'elles una funcionalitat a l'aplicació Android. Per altra banda estan les classes que no són públiques a través de la xarxa i s'encarreguen de gestionar la base de dades i de transformar les respostes de la base de dades a format JSON.

Totes les classes públiques a través de la xarxa són bastant similars, per tant es mostrarà a tall d'exemple la classe Login, encarregada de proporcionar la funcionalitat d'iniciar sessió al telèfon amb un usuari. També es mostrarà el mètode corresponent al login de la classe gestora de la base de dades així com el mètode encarregat de transformar la resposta de la base de dades a format JSON.

En la classe login es pot observar la URI definida mitjançant les anotacions @Path, el mètode que s'utilitzarà per accedir al recurs (@GET), i el tipus de retorn que produirà la consulta del client (@Produces(MediaType.APPLICATION_JSON)).

```
package com.biblioteca.jersey;

import javax.ws.rs.GET;
import javax.ws.rs.Path;
import javax.ws.rs.Produces;
import javax.ws.rs.QueryParam;
import javax.ws.rs.core.MediaType;

//Classe que s'utilitza per a realitzar el login amb un usuari
@Path("/login")
public class Login {

    @GET
    @Path("/autentifica")
    @Produces(MediaType.APPLICATION_JSON)
    //S'indica que es vol utilitzar el metode Get, la URI amb la qual es pot accedir al servei
    // i els parametres esperats
    //http://localhost:8080/biblioteca/login/autentifica?usuari=aquerol&password=1234
    public String autentifica(@QueryParam("usuari") String usuari,
        @QueryParam("password") String password)
    {
        String resposta="";
        //Es comprova que els parametres obtinguts del client Android no siguin null
        if (ConvertorsJSON.stringNotNull(usuari) && ConvertorsJSON.stringNotNull(password)) {

            try {
                //Es fa la petició a la base de dades
                resposta = ConnexioBDD.login(usuari, password);

            } catch (Exception e) {
                e.printStackTrace();
            }
        }
        //Es retorna la resposta ja en format JSON provinent de la classe gestora de la base de dades
        return resposta;
    }
}
```

Es mostren els mètodes obrirConnexio i login de la classe ConnexioBDD:

```
public class ConnexioBDD {

    /* Metode per a obrir una connexio amb la base de dades MySQL
       Retorna una connexió amb la base de dades
       Si la connexió es correcta retorna la connexio
       Si la connexió no es correcta retorna null*/
    @SuppressWarnings("finally")
    public static Connection obrirConnexio() throws Exception {
        Connection con = null;
        try {
            Class.forName("com.mysql.jdbc.Driver");
            con = DriverManager.getConnection("jdbc:mysql://localhost:3306/biblioteca","root",
                "Xancla90");
        } catch (Exception e) {
            throw e;
        } finally {
            return con;
        }
    }

    /*
       Metode que realitza el login a l'aplicació
       Retorna un objecte Usuari en format JSON
       Si el login es correcte l'objecte usuari conte el nom i el id de l'usuari
       Si el login no es correcte l'objecte usuari conte el nom="" i el id=-1
    */
    public static String login(String usuari, String password) throws Exception {

        Usuari usuariRet = new Usuari();
        Connection connexioBD = null;

        try {
            //S'obre la connexió amb la base de dades i es prepara la consulta
            connexioBD = ConnexioBDD.obrirConnexio();
            Statement statement = connexioBD.createStatement();
            String query = "SELECT id, NomUsuari FROM usuaris WHERE NomUsuari = '" +usuari+ "'
                AND Contrasenya=" + "'" + password + "'";

            //S'executa la consulta
            ResultSet resultat = statement.executeQuery(query);

            //Si hi ha un resultat vol dir que l'usuari i el password son correctes
            if (resultat.next()) {
                usuariRet.setId(resultat.getInt(1));
                usuariRet.setNomUsuari(resultat.getString(2));
            }
            //Ens asegurem de tancar la connexió amb la base de dades si alguna cosa surt malament
        } catch (Exception e) {
            if (connexioBD != null) {
                connexioBD.close();
            }
            throw e;
        } finally {
            if (connexioBD != null) {
                connexioBD.close();
            }
        }
        //Es retorna el resultat en format JSON
        return ConversorsJSON.usuariToJSON(usuariRet).toString();
    }
    (...)
}
```

Finalment es mostra el mètode `usuariToJSON` de la classe `convertorsJSON`, el qual s'encarrega de convertir la resposta de la base de dades a format JSON:

```
public class ConvertorsJSON {
    (...)

    //metode que transforma la informació d'un usuari a JSON
    public static JSONObject usuariToJSON(Usuari usuari){
        JSONObject ret = new JSONObject();

        try{
            ret.put("id", usuari.getId());
            ret.put("nomUsuari", usuari.getNomUsuari());

        }catch(JSONException e){
            System.err.println("Error ConvertorsJSON.UsuariToJSON: Error al crear Usuari en JSON");
        }
        return(ret);
    }
    (...)
}
```

4.5- Anàlisi de l'estat del projecte

Considero que un cop finalitzat el projecte el resultat és bastant complet, ja que es proporcionen totes les funcionalitats previstes inicialment, a més de diverses d'extra mitjançant l'aplicació bibliotecari.

No obstant es podrien dur a terme múltiples millores:

- Incorporar una capa de seguretat entre la connexió del client Android i el servidor mitjançant HTTPS. Actualment les dades s'envien en text pla juntament amb la URL d'accés al servei, el qual és un forat de seguretat molt gran.
- Incorporar filtres als EditText per tal que els usuaris no puguin introduir caràcters no desitjats a l'aplicació.
- Incorporar un sistema de notificacions per als recordatoris de retorn i recollida de llibres.
- Millorar la interfície gràfica i desenvolupar una icona per a l'aplicació.
- Incorporar algunes de les millores suggerides pels usuaris en les entrevistes en profunditat o bé algun dels requisits opcionals definits en l'apartat 1.2 de la memòria.
- Millorar l'algorisme de cerca de llibres i oferir opcions més avançades per filtrar la cerca com per exemple per any de publicació o popularitat.
- Incorporar un sistema per a que el bibliotecari pugui donar d'alta i baixa llibres sense haver d'editar-los manualment a la base de dades.

Tot i que són moltes les millores que es poden dur a terme al projecte, considero que s'han assolit els objectius plantejats inicialment, per tant es dona el projecte per finalitzat amb data 14/06/2017.

5 - Proves

5.1- Com es realitzaran les proves

Les proves de l'aplicació s'han dut a terme tant des de l'emulador d'Android Studio com amb un telèfon físic, un Xiaomi Mi4W (Versió d'Android 6.0.1).

Per tal de realitzar les proves es va descartar les proves d'unitat local, ja que únicament es poden executar en classes que no tenen dependències amb l'API d'Android i en el cas d'aquest projecte les úniques classes en què es podrien executar aquests tests són bastant simples i no té molt sentit realitzar un test en elles.

Inicialment es tenia previst realitzar proves instrumentals utilitzant els Frameworks JUnit i Espresso, els quals permeten realitzar tests juntament amb l'API d'Android. No obstant a l'hora de desenvolupar els tests es va detectar que s'invertia més temps depurant i comprovant que el codi del propi test fos correcte que no pas realitzant proves útils.

Per tant es va decidir realitzar únicament dos test mitjançant Espresso i JUnit per comprovar com era el funcionament d'aquest Framework i a continuació es van realitzar proves manuals de totes les funcionalitats mitjançant l'emulador d'Android Studio i el telèfon físic mencionat anteriorment. Es mostraran algunes de les proves realitzades corresponents als casos d'ús definits a l'apartat 3.2 de la memòria.

5.2 - Proves amb Espresso i JUnit

A continuació es mostren les dues proves realitzades mitjançant Espresso i JUnit:

Primer test

Es comprova que els paràmetres de cerca es transmeten correctament de l'activitat BuscadorLlibres a l'activitat CatalegLlibres.

```
@LargeTest
@RunWith(AndroidJUnit4.class)

public class testParametresCerca {

    @Rule
    public IntentsTestRule <BuscadorLlibres> buscadorLlibres =
        new IntentsTestRule<>(BuscadorLlibres.class);

    @Test
    public void verificarParametresCerca() {
```

```

//Introduim que volem buscar la paraula clau informatica al editText
paraulesClau
onView(withId(R.id.paraulesClau)).perform(typeText("informatica"),closeSoftKey
board());

//Indiquem que volem seleccionar el tema "informatica" de la llista de temes
desplegables
onView(withId(R.id.llista_temes)).perform(click());
onData(allOf(is(instanceOf(String.class)))) .atPosition(1).perform(click());

//Fem click al boto de Buscar
onView(withId(R.id.bt_fes_cerca)).perform(click());

//Comprovem que al Cataleg de llibres s'ha rebut les paraules clau indicades
intended(allOf(
    hasExtras(allOf(
        hasEntry(equalTo("tema"), equalTo("Informatica"))
        ,hasEntry(equalTo("paraulesClau"), equalTo("informatica"))))
    ,toPackage("com.tfg.andreu.biblioteca")));
}
}

```

Segon test

Es comprova que els framework funcionen correctament realitzant un test molt senzill, en aquest cas es comprova que el paquet de l'aplicació sigui el correcte.

```

@RunWith(AndroidJUnit4.class)
public class testComprovarConfiguracioOK {

    @Test
    public void useAppContext() throws Exception {

        //Test molt senzill per comprovar que els test s'executen correctament
        Context appContext = InstrumentationRegistry.getTargetContext();
        assertEquals("com.tfg.andreu.biblioteca", appContext.getPackageName());

    }
}

```

5.3 - Proves amb l'emulador Android

Tot i que s'han realitzat moltes més proves, per tal de no carregar massa la memòria a continuació es mostra una de les proves realitzades mitjançant l'emulador Android, aquesta prova es correspon al cas d'ús CU-003 definit a l'apartat 3.2 de la memòria, en aquest cas d'ús es vol reservar un llibre del catàleg de la biblioteca.

S'ha escollit aquest cas d'ús perquè també inclou el CU-001 en el qual l'usuari vol buscar la informació d'un llibre al catàleg de la biblioteca i el cas d'ús CU-002 en el qual l'usuari inicia sessió a l'aplicació.

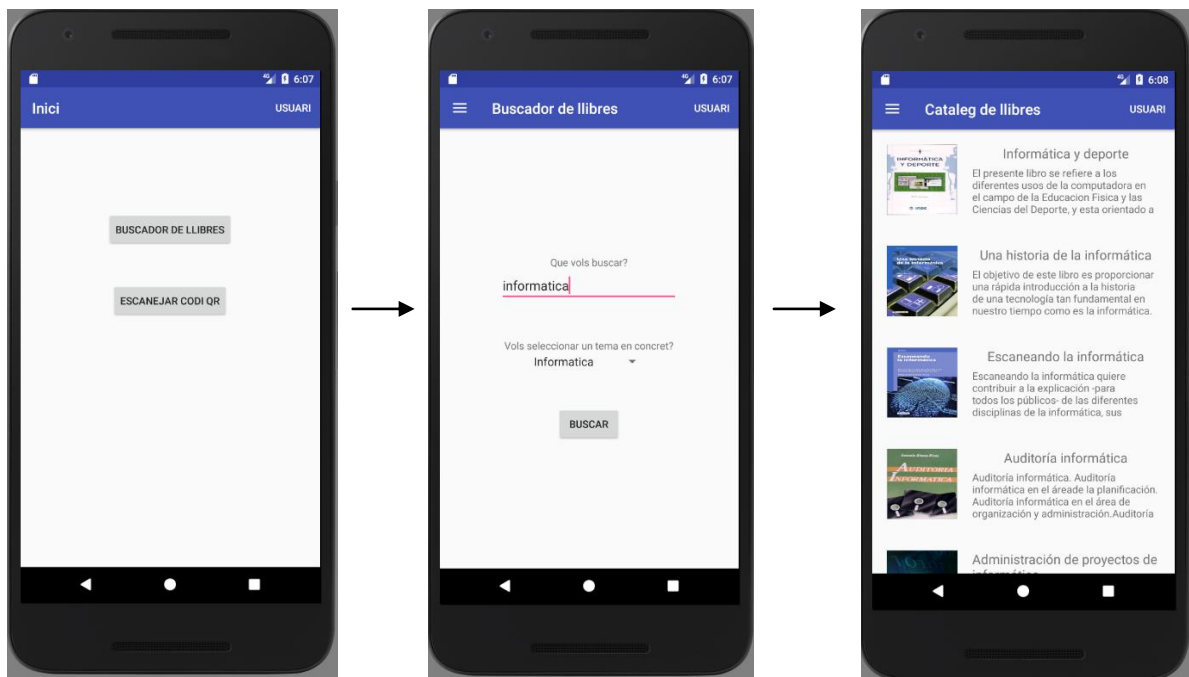
Prova CU-003:

Passos realitzats:

1) L'usuari busca un llibre al catàleg (CU-001) i el selecciona per obtenir més informació.

CU-001:

- 1.1) L'usuari selecciona l'opció "Buscador de llibres".
- 1.2) El sistema mostra la pantalla per a buscar llibres.
- 1.3) L'usuari introdueix una paraula clau i selecciona un tema.
- 1.4) El sistema mostra un llistat de llibres del tema seleccionat per l'usuari i amb el títol que contingui la paraula clau seleccionada.



2) El sistema mostra la informació del llibre seleccionat.



3) L'usuari reserva el llibre.

3.a) Si l'usuari no ha iniciat sessió al sistema.

3.a.1) El sistema indica a l'usuari que ha d'iniciar sessió.

3.a.2) L'usuari inicia sessió al sistema (CU-002).

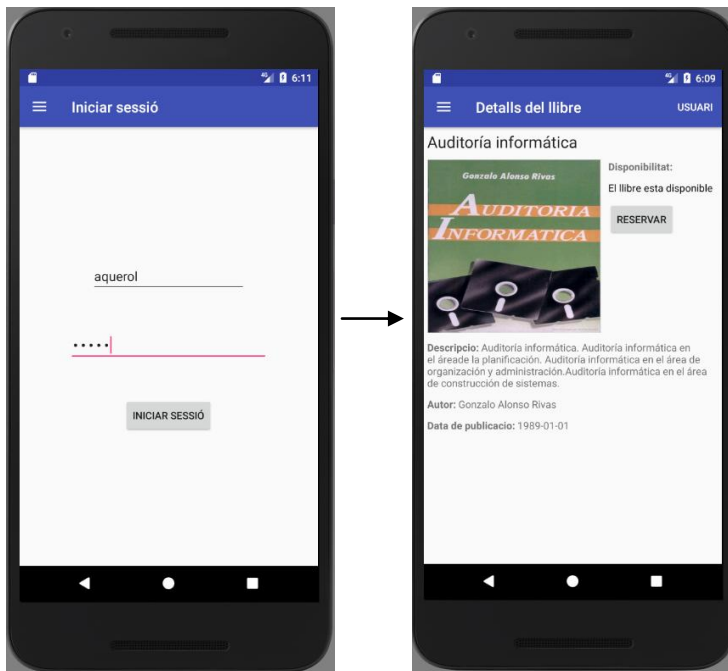
CU-002:

3.a.2.1) L'usuari introdueix el seu nom d'usuari i contrasenya.

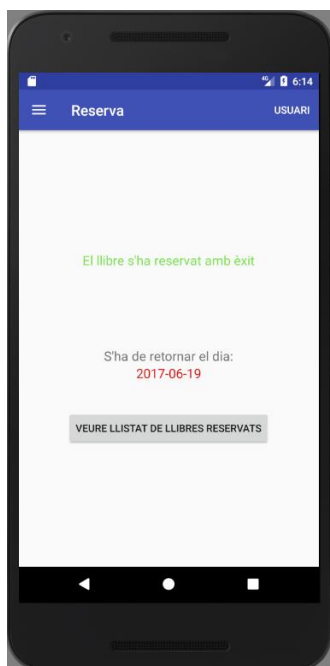
3.a.2.2) El sistema comprova que el nom d'usuari i la contrasenya concordin amb algun usuari registrat a l'aplicació.

3.a.2.3) El sistema inicia sessió amb l'usuari indicat i guarda la informació de l'usuari al telèfon.

3.a.3) Es torna al punt 2.



4) El sistema mostra la confirmació de la reserva juntament amb la data en què haurà de ser retornat el llibre



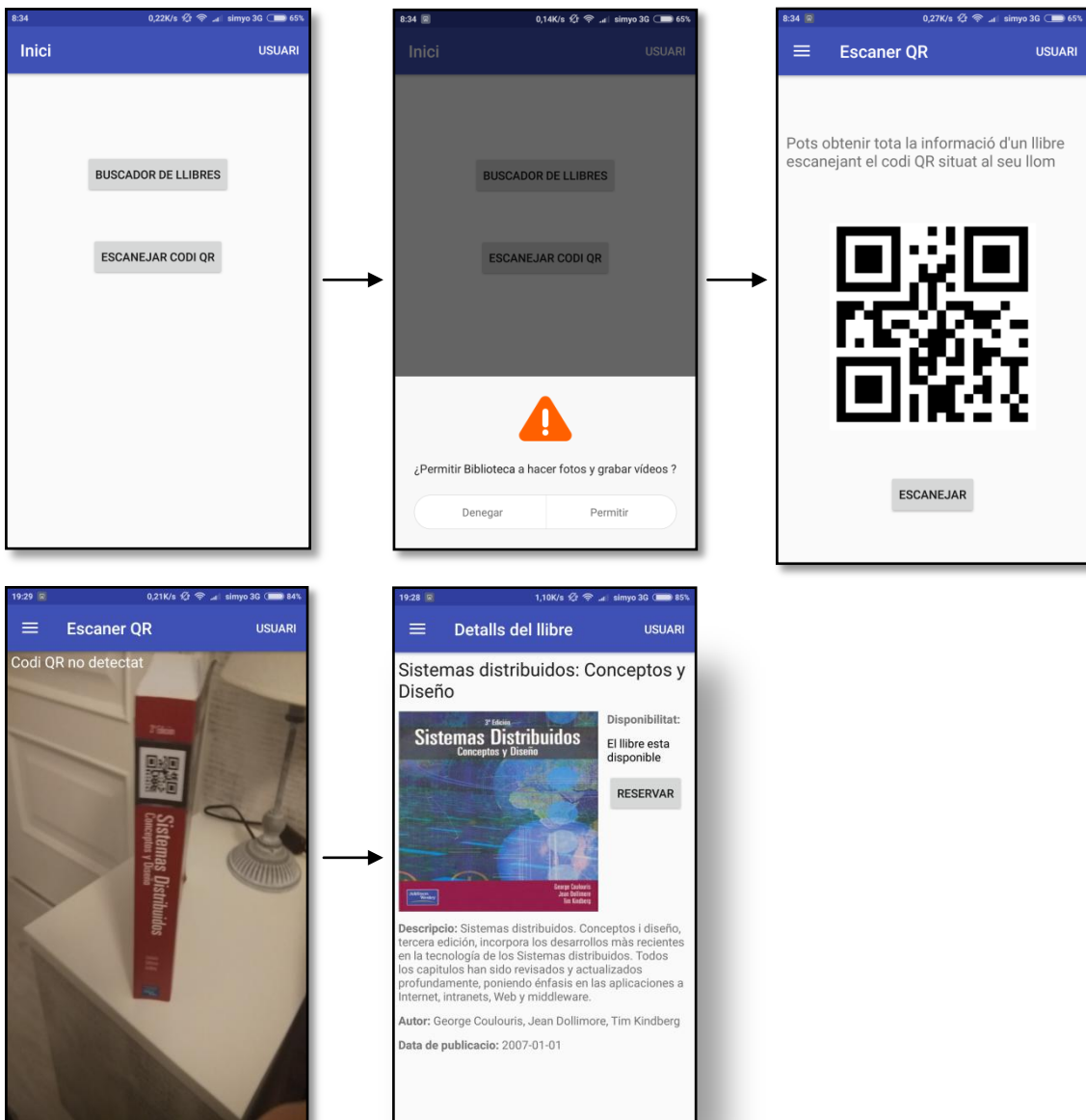
5.4 - Proves amb el telèfon físic

Tal i com s'ha comentat anteriorment les proves s'han realitzat amb un telèfon Xiaomi Mi4W (Versió d'Android 6.0.1). En aquest cas es mostrarà les proves corresponents al cas d'ús CU-004 i CU-006.

Prova CU-004

Passos realitzats:

- 1) L'usuari selecciona l'opció "Escanejar codi QR".
- 2) El sistema pregunta a l'usuari si vol permetre que l'aplicació pugui fer fotos i gravar vídeos.
- 3) Si l'usuari accepta, el sistema mostra la pantalla amb les instruccions per escanejar el codi.
- 4) L'usuari escaneja el codi QR.
- 5) El sistema mostra la pantalla amb la informació del llibre identificat pel codi QR.



Prova CU-006

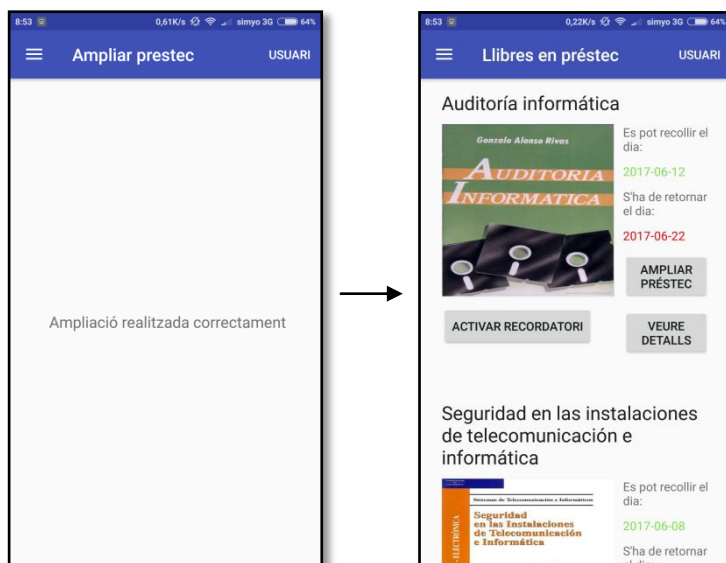
En aquest cas s'ometrà el procés d'iniciar sessió ja que ja ha estat mostrat en la prova del cas d'ús CU-003.

Passos realitzats:

- 1) L'usuari consulta el llistat de llibres reservats (CU-005).
 - 1.1) L'usuari selecciona l'opció "Llibres que tens en préstec".
 - 1.2) El sistema mostra la pantalla amb el llistat de llibres en préstec de l'usuari.



- 2) L'usuari selecciona l'opció d'ampliar préstec d'un llibre del llistat.
- 3) El sistema amplia el temps de préstec del llibre, actualitza la data en què ha de ser retornat.



6 - Conclusió

Dur a terme aquest projecte ha estat tot un repte, he pogut experimentar de primera mà com és el desenvolupament d'un projecte partint de l'inici conceptual fins a la implementació i realització de proves finals.

Ha estat una experiència difícil però molt enriquidora, en la qual en diverses ocasions he tingut la sensació de no saber com avançar, principalment perquè Android era una tecnologia completament nova per a mi, i tal i com va avisar el consultor les primeres setmanes del projecte, té una corba d'aprenentatge bastant elevada.

No obstant estic satisfet amb el resultat obtingut, considero que he complert el que per mi és l'objectiu principal del projecte a l'adquirir una base bastant sòlida d'Android. A més també he aconseguit complir els altres objectius definits en lliurar una aplicació completa i funcional, tot i que tal com s'ha comentat anteriorment amb un ampli marge de millora.

El gran ventall de llibreries que disposa Android pot ser aclaparador en un principi, tot i així ha estat molt interessant aprendre a fer servir les API utilitzades en aquest projecte, estic segur que aquests coneixements em seran molt útils si en un futur la meua carrera professional acaba dedicada a la creació d'aplicacions per aquesta plataforma tan estesa avui en dia.

Sóc de l'opinió que Android com a plataforma per al desenvolupament d'aplicacions té molt futur, tant per a dispositius mòbils com per altres tipus d'aparells. El fet d'haver realitzat aquest projecte m'ha permès tenir una primera presa de contacte amb aquesta tecnologia, gràcies a la qual podré encarar en un futur la creació d'altres aplicacions Android amb una solidesa i una experiència molt valuoses.

Finalment també voldria comentar que estic bastant satisfet d'haver decidit implementar una petita aplicació per al bibliotecari, la qual considero que arrodoneix el treball realitzat i permet oferir un projecte molt més complet.

7 - Bibliografia

Smartphone OS Market Share, 2016 Q3 [en línia]. IDC. [Data de consulta: 4 de març de 2017]
<<http://www.idc.com/promo/smartphone-market-share/os>>

Android Tutorial [en línia]. [Data de consulta: Any 2017 Múltiples consultes al llarg del projecte]
<<http://www.tutorialspoint.com/android/index.htm>>

Developing Android Apps by Google [en línia]. Google. [Data de consulta: Any 2017 Múltiples consultes al llarg del projecte]
<<https://www.udacity.com/course/new-android-fundamentals--ud851>>

Android Asynchronous Http Client [en línia]. James Smith. [Data de consulta: Any 2017 Múltiples consultes al llarg del projecte]
<<http://loopj.com/android-async-http/>>

Tutorial de uso de Android Asynchronous HTTP Client (LOOPJ) [en línia]. Angel Manuel Gamaza Domínguez, José Miguel Otte Sainz-Aguirre. [Data de consulta: Any 2017 Múltiples consultes al llarg del projecte]
<http://rodin.uca.es/xmlui/bitstream/handle/10498/18591/LOOPJ_Gamaza_Otte.pdf?sequence=1>

Fòrum stackoverflow.com múltiples pàgines [60en línia]. . [Data de consulta: Any 2017 Múltiples consultes al llarg del projecte]
<<https://stackoverflow.com/>>

Entrenament per a desenvolupadors Android [en línia]. Google. [Data de consulta: Any 2017 Múltiples consultes al llarg del projecte]
<<https://developer.android.com/training/index.html>>

Introduction to Glide [en línia]. nuuneoi. [Data de consulta: Any 2017 Múltiples consultes al llarg del projecte]
<<https://inthecheesefactory.com/blog/get-to-know-glide-recommended-by-google/en> >

Jersey RESTful Web Services in Java [en línia]. . [Data de consulta: Any 2017 Múltiples consultes al llarg del projecte]
<<https://jersey.github.io/>>

QR code generator [en línia]. Denso wave incorporated . [Data de consulta: Any 2017 Múltiples consultes al llarg del projecte]
<<http://es.qr-code-generator.com/>>

8 - Glossari

- QR: Codi de resposta ràpida, és l'evolució dels codis de barres, s'utilitza per identificar els llibres de la biblioteca.
- RestFul: Sistema que segueix els principis de l'arquitectura REST (transferència d'estat representacional)
- App : Aplicació per a dispositius mòbils.
- UML: Llenguatge unificat de modelatge, és un llenguatge gràfic per modelar sistemes.
- Nested class: Classe que està implementada dintre del cos d'una altra classe.
- XML: Llenguatge de marques extensible, és un llenguatge de marques que permet guardar dades en un format llegible tant pels humans com pels ordenadors.
- Layout: Distribució dels elements gràfics d'una aplicació.
- JSON: JavaScript Object Notation, és un format per representar informació en cadenes de text.
- IDE: Entorn de desenvolupament integrat, és un programa que serveix per desenvolupar altres programes.
- Framework: Entorn de treball que proporciona funcionalitats per a facilitar la programació.
- Open source: codi obert, és una llicència per distribuir codi lliure.
- Thread: fil d'execució en un sistema informàtic.
- API: Interfície de programació d'aplicacions, és un conjunt de classes i mètodes d'una llibreria els quals estan pensats per proporcionar funcionalitats ja implementades a una nova aplicació.
- Widget: Element de la interfície gràfica.
- Login: Iniciar sessió a l'aplicació.
- Logout: Tancar sessió a l'aplicació.
- id: Numero identificatiu d'un usuari o llibre.

9 - Annex

9.1- Aplicació Bibliotecari

Un cop ja tenia pràcticament acabada l'aplicació principal del projecte vaig tenir la sensació que l'aplicació estava poc arrodonida i que presentaria un projecte com "a mitges", ja que tot i proporcionar les funcionalitats descrites en els apartats anteriors de la memòria, realment no tenia cap funcionalitat amb la qual prestar un llibre a un usuari o tornar un llibre prestat a la biblioteca, a més tampoc estava incorporant l'alta d'usuaris.

Per tant vaig pensar que el que tenia més sentit era crear una petita aplicació com annex al projecte per al bibliotecari de la biblioteca, amb aquesta aplicació el bibliotecari pot prestar un llibre a un usuari, tornar un llibre a la biblioteca i donar d'alta usuaris, d'aquesta forma podia seguir per una banda amb la idea exposada al punt 4.2.3 i per altra banda podia incorporar informació més valuosa a la pantalla inicial (Home) tal com suggeria el consultor de disseny en una de les valoracions del projecte.

L'aplicació Bibliotecari està formada per cinc activitats:

Activitat Home

És l'activitat inicial de l'aplicació, es pot accedir a totes les funcionalitats de l'aplicació a través d'aquesta pantalla, no s'ha inclòs menú de navegació a cap de les activitats a causa de que al ser una aplicació amb tan poques pantalles he considerat que no era necessari de moment.

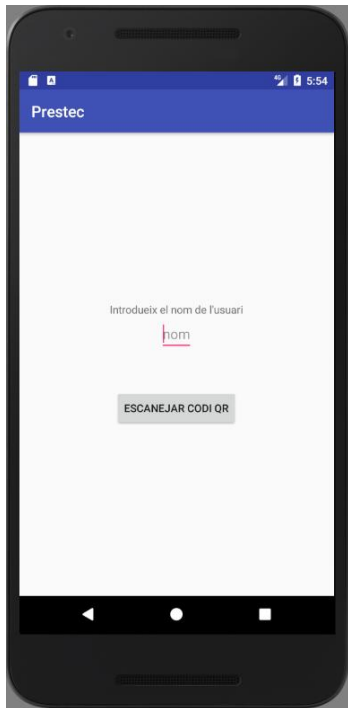


Les opcions disponibles són les següents:

- *Préstec d'un llibre:* Permet al bibliotecari prestar un llibre de la biblioteca a un usuari.
- *Devolució d'un llibre:* Permet al bibliotecari tornar un llibre prestat a la biblioteca.
- *Donar d'alta un usuari:* Permet crear un nou usuari al sistema.

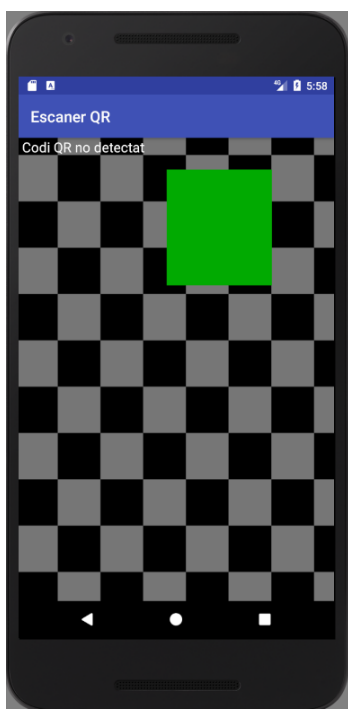
Activitat PrestarLlibreUsuari

És l'activitat a la qual es viatja en seleccionar l'opció "Préstec d'un llibre" en la pantalla inicial. Aquesta activitat s'encarrega de recollir el nom de l'usuari al qual se li està prestant el llibre, d'aquesta forma es pot passar a la següent activitat, la lectora de codis QR i és aquesta última la que s'encarrega d'enviar aquesta informació al servidor.



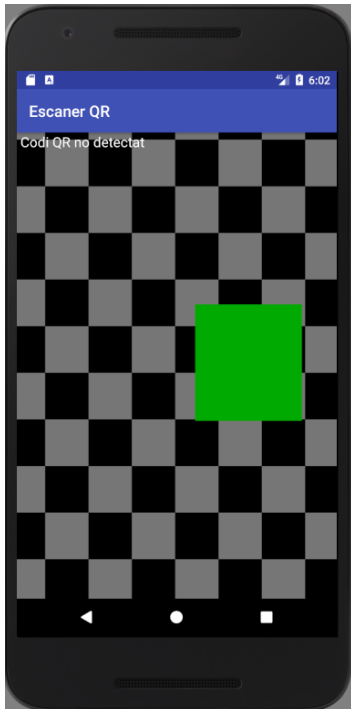
Activitat PrestarLlibreQR

Es l'activitat encarregada de llegir el codi QR del llibre i un cop llegit invocar al servidor amb el id del llibre i el nom d'usuari passat per paràmetre de l'activitat anterior. Al igual que amb l'aplicació Biblioteca no es pot accedir a aquesta activitat si l'usuari no dona permís a l'aplicació per utilitzar la càmera.



Activitat TornarLlibreQR

És l'activitat encarregada de llegir el codi QR d'un llibre i tornar-lo a la biblioteca, igual que en l'activitat anterior s'encarrega d'invocar al servidor i no es pot accedir a ella sense permís de l'usuari. En aquest cas no es pregunta pel nom de l'usuari que està retornant el llibre perquè un llibre únicament pot estar prestat a un usuari en un moment determinat, i com abans de poder tornar el llibre l'usuari l'haurà d'haver prestat el sistema ja coneixerà, en cas que fos necessari, quin és l'usuari que està retornant el llibre.

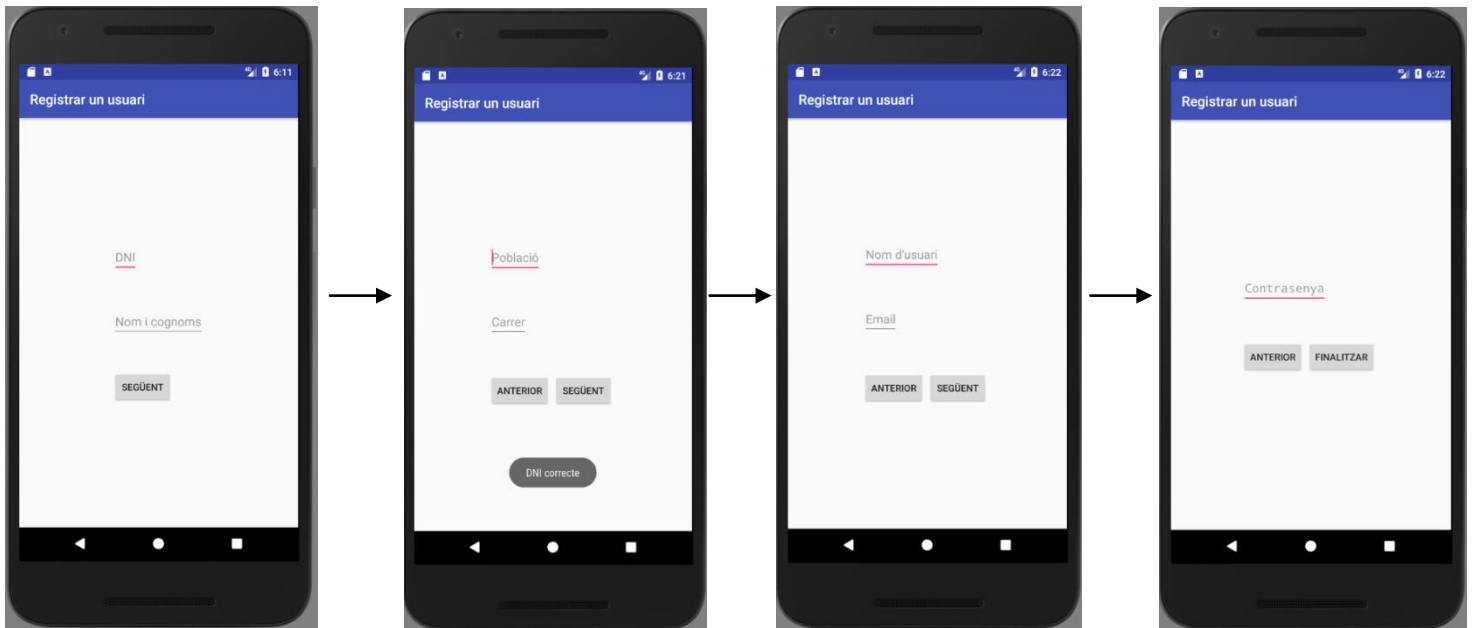


Activitat RegistrarUsuari

Es l'activitat encarregada de recollir les dades d'un nou usuari i invocar al servidor per a crear-lo al sistema. Actualment es confia que les dades introduïdes per part del bibliotecari seran correctes, ja que es realitzen molt poques comprovacions per determinar si les dades son correctes o no. Per exemple es comprova si el DNI o el nom d'usuari no existeixen a la base de dades, no obstant no es segueix cap algoritme per determinar si un DNI es vàlid comprovant el numero amb la seva lletra de control.

Aquesta activitat te una particularitat respecte a totes les altres del projecte, ja que utilitza un tipus de widget ViewPager per a mostrar quatre layouts diferents (registrar 1 - 2 - 3 - 4) . Es va decidir utilitzar aquest widget perquè considero que el procediment per donar d'alta un usuari es molt mes entenedor i còmode si es divideix la introducció de dades en múltiples pantalles.

En les imatges següents es pot veure el widget ViewPager contenint tots els layouts de registre, per a navegar d'un layout al següent es fan servir els botons Anterior i Següent.



Per part del servidor, s'ha creat les següents classes encarregades de proporcionar les funcionalitats necessàries a l'aplicació:

- PrestarLlibre: Classe que s'encarrega de comunicar-se amb la classe gestora de la base de dades per a prestar un llibre a un usuari en concret passat per paràmetre.
- TornarLlibre: Classe que s'encarrega de comunicar-se amb la classe gestora de la base de dades per a tornar un llibre a la biblioteca.
- ComprovarDNI: Classe que és invocada en el moment que el bibliotecari introdueix un DNI en el procediment de creació d'un nou usuari, s'encarrega de comunicar-se amb la classe gestora de la base de dades per a comprovar que el DNI introduït no correspongui al de cap usuari ja existent al sistema.
- ComprovarNomUsuari: Classe que és invocada en el moment que el bibliotecari introdueix un nom d'usuari en el procediment de creació d'un nou usuari, s'encarrega de comunicar-se amb la classe gestora de la base de dades per a comprovar que el nom d'usuari introduït no correspongui al de cap usuari ja existent al sistema.
- CrearUsuari: Classe que s'encarrega de comunicar-se amb la classe gestora de la base de dades per a crear un nou usuari al sistema amb tots els paràmetres recollits per l'aplicació bibliotecari.

9.2- Comentaris

9.2.1 - Usuaris per defecte:

Per defecte el servidor te creats els següents usuaris de prova amb els quals es pot iniciar sessió a l'aplicació:

Usuari	Contrasenya
aquerol	12345
ssole	12345
jquerol	12345
cgonzalez	12345

9.2.2 - Codis QR vàlids:

Els codis QR vàlids per a l'aplicació són simplement els números identificatius dels llibres, per tant si es vol provar la funcionalitat per a escanejar codis QR el que s'ha de fer és generar un codi QR amb alguna eina, com per exemple <http://es.qr-code-generator.com/>, i introduir com a codi únicament el número identificatiu del llibre. Per exemple: 1

Actualment hi ha 20 llibres a la base de dades per tant els números identificatius vàlids van del número 1 fins al 20.

9.2.3 - Informació dels llibres.

Tant la informació dels llibres com les imatges de les seves portades han estat extretes del buscador [Google Books](#).