

**5640-1**  
**APLICACIÓ ANDROID DE GESTIÓ**  
**D'UN CLUB DE BÀSQUET**

Memòria del Projecte Fi de Carrera  
d'Enginyeria en Informàtica

Realitzat per Marc Casanova Barjuan  
i dirigit per Ramon Grau Sala

Bellaterra, 2 de setembre de 2014





El sotasignat Ramon Grau Sala

Professor/a de l'Escola d'Enginyeria de la UAB,

**CERTIFICA:**

Que el treball a què correspon aquesta memòria ha estat realitzat sota la seva direcció per en **Marc Casanova Barjuan**

I per tal que consti firma la present.

Signat Ramon Grau Sala

Bellaterra, 2 de Setembre de 2014



# Agraïments

Agrair primerament a la meva família el suport que m'ha donat des del principi i també com m'ha anat tornant a guiar cap al camí quan em desviada sense adonar-me'n, i no veia gens clar el resultat final.

També voldria agrair-li a la meva parella els ànims que m'ha donat en moments difícils i l'entrega en la correcció dels possibles errors en l'escriptura de la memòria.

Vull agrair en especial als caps de l'empresa on treballo, que sense haver-ho de demanar m'han donat tot el suport a l'hora de reduir la meva jornada laboral per a poder tenir a punt aquest projecte.

Vull agrair al meu tutor Ramon Grau la confiança i la llibertat que m'ha proporcionat al realitzar un projecte d'aquesta magnitud, creient sempre en el meu potencial.

A tots vosaltres, gràcies.



## **Resum**

El següent document correspon a la memòria del projecte fi de carrera d'Enginyeria Informàtica en què hi podem trobar l'explicació del disseny i la implementació de l'aplicació següent: gestió d'un club de bàsquet sobre la plataforma Android.

Aquesta memòria correspon a un projecte desenvolupat per una persona en solitari. S'hi exposen tres parts clarament diferenciades: la justificació de l'elecció d'Android, el disseny i la implementació de l'aplicació. La implementació conté funcions com ara la pissarra per a l'entrenador, jugades interactives i la creació d'actes digitals .

## **Resumen**

El siguiente documento corresponde a la memoria del proyecto de fin de carrera de Ingeniería Informática, donde se explica el diseño e implementación de la aplicación siguiente: gestión de un club de básquet sobre la plataforma Android.

Esta memoria corresponde a un proyecto desarrollado por una persona en solitario. Se exponen tres partes diferenciadas: la justificación de la elección de Android, el diseño y la implementación de la aplicación. La implementación contiene funciones como la pizarra para el entrenador, jugadas interactivas o la creación de actas digitales.

## **Abstract**

The following document deals with the report of the final Information Technologies degree, which contains the design and implementation of the following application: management of a basketball club on an Android platform.

This belongs to a project developed by one person. In this report you will find three basic parts: the justification of chose of the Android platform, the design and the app implementation. The implementation contains functions such as the trainer's blackboard, interactive plays or creation of digital match reports.





# ÍNDEX

<b>1. INTRODUCCIÓ</b> .....	1
1.1 Objectiu/s del projecte .....	1
1.2 Breu introducció a l'estat de l'art del tema proposat .....	2
1.3 Estudi de viabilitat del projecte .....	3
1.4 Eines utilitzades.....	4
1.5 Planificació temporal del treball.....	5
<b>2. ESTAT DE L'ART</b> .....	7
2.1 Dispositius mòbils .....	7
2.1.1 Introducció.....	7
2.1.2 SmartPhone/Tablet .....	8
2.2 Sistemes Operatius.....	11
2.2.1 Android.....	11
2.2.2 iOS .....	12
2.2.3 Windows Phone.....	13
2.2.4 BlackBerry OS .....	14
2.3 Justificació de l'elecció .....	15
<b>3. ENTORN DE TREBALL</b> .....	20
3.1 Android .....	20
3.2 Unity .....	23
3.3 SQL Developer .....	27
3.4 Anàlisi de requeriments.....	29
<b>4. DISSENY</b> .....	32
4.1 Diagrama detallat.....	32
4.2 Interfície d'usuari.....	34
4.2.1 Pantalla de logueig/registre .....	34
4.2.2 Pantalla de càrrega.....	35
4.2.3 Pantalla de menú principal .....	36
4.2.4 Pantalla de pissarra .....	37
4.2.5 Pantalla d'acta.....	38
4.2.6 Pantalla d'estadística.....	40
4.2.7 Pantalla de tàctica.....	42
4.2.7.1 Pantalla d'enregistrar jugada.....	43
4.2.7.2 Pantalla de reproduir jugada .....	44
4.2.7.3 Pantalla de simulació de defenses .....	45
4.2.7.4 Pantalla de manager de jugades .....	46
<b>5. IMPLEMENTACIÓ</b> .....	47
5.1 Base de Dades SQL.....	47
5.1.1 Creació de la base de dades.....	47
5.1.2 Creació de php's per a la consulta.....	49
5.2 Android .....	51
5.2.1 Pantalla de logueix .....	51
5.2.2 Pantalla de càrrega.....	53

5.2.3	Pantalla principal.....	54
5.2.4	Pantalla de pissarra .....	56
5.2.5	Pantalla d'acta.....	63
5.2.6	Pantalla d'estadística.....	71
5.2.7	Mòdul SQLite.....	73
5.2.8	Mòdul Anàlisi .....	74
5.3	Unity .....	75
5.3.1	Pantalla principal.....	76
5.3.2	Pantalla de l'editor .....	77
5.3.3	Pantalla de reproducció.....	80
5.3.4	Pantalla de reproducció 3D (no inclosa) .....	81
5.3.5	Joc Encistella'm .....	83
<b>6.</b>	<b>CONCLUSIONS.....</b>	<b>84</b>
6.1	Conclusió.....	84
6.2	Valoració personal.....	85
6.3	Possibles millores .....	86
6.4	Problemes.....	87
<b>7.</b>	<b>REFERÈNCIES I BIBLIOGRAFIA.....</b>	<b>88</b>



# 1. INTRODUCCIÓ

## 1.1 Objectiu/s del projecte

Avui en dia, les necessitats d'un club de bàsquet, tant sigui un club petit com un de gran, passen per tenir certes necessitats de gestió dels seus equips d'una forma diferent.

El fet de tenir eines avançades com telèfons mòbils o tablettes ens proporciona un gran ventall de possibilitats alhora de distribuir/unificar tota la informació que aquest equip o club pot generar, podent exportar tota aquesta informació al moment, i distribuir-la per als altres integrants de la xarxa.

L'evolució del software ha fet que actualment sigui possible construir aplicacions que permetin la consulta i generació de la informació necessària, tant siguin els resultats dels equips, com estadístiques de jugadors, consulta o construcció de jugades.

Aquest projecte té com a objectius:

- Investigació de les tècniques i eines modernes que s'utilitzen per al desenvolupament d'una aplicació per al sistema operatiu.
- Implementació de tots els mòduls necessaris per a la construcció d'una interfície adequada per als usuaris.
- Gestionar tota la informació del club en una Base de Dades per a la seva correcta sincronització amb tots els dispositius que usin aquesta aplicació.
- Realitzar una interfície de simulacions de jugades, i simulacions de defenses en 3D amb el motor gràfic Unity.

## **1.2 Breu introducció a l'estat de l'art del tema proposat**

Els clubs de bàsquet actualment disposen, en la seva gran majoria, únicament d'una pàgina web senzilla on mostren la informació referent al club com ara: el nom i disposició dels jugadors de cada equip i categoria, els colors de l'equip, l'adreça del pavelló, una zona de galeria de fotos, etc.

El que em cridava l'atenció era que els clubs no disposessin d'una aplicació mòbil on es pogués consultar la informació referent als partits (actes), resultats de totes les categories dins del club, adreces dels pavellons on es va a jugar, horaris dels partits de cada equip, i en general, de tota la informació actualitzada de cada jornada en una aplicació senzilla i pràctica que faci la vida més fàcil a tots els amants del món del bàsquet.

A més a més, des de la coordinació dels clubs de bàsquet s'acostuma a proporcionar als entrenadors i al personal que gestiona els diferents equips tàctiques, tècniques i hàbits de joc que des de l'escola del Club es volen instaurar en els jugadors; tot això seria més fàcil proporcionar-ho des d'una sola plataforma on tots aquests integrants hi poguessin accedir. Des d'aquesta plataforma gestionada directament per la direcció del club també es potenciaria un creixement esportiu dels jugadors més òptim, facilitant la consulta de jugades d'atac, tàctiques de defensa, tècnica individual, exercicis de pista, simulació de defenses segons l'atac que es faci, exercicis d'entrenament. No cal dir també, que poder posar tota aquesta informació en conjunt i amb un accés senzill, facilitaria no només el creixement dels jugadors, sinó que també la tasca dels entrenadors.

## 1.3 Estudi de viabilitat del projecte

Per poder dur a terme aquest projecte, he realitzat un estudi de viabilitat que servirà per a determinar si és un projecte realitzable i si es poden complir tots els objectius perseguits.

Per poder dur a terme el projecte, tots els aspectes tècnics necessaris estan coberts, ja que el club elegit "Club bàsquet la Garriga" disposa de diferents equips de diferents categories, amb gran quantitat de jugadors, entrenadors amb dispositius mòbils per a poder fer la consulta i delegats de cada equip per a poder fer les actes de cada partit al moment.

Les eines de desenvolupament necessàries estan llistades en el proper apartat. Són les eines de desenvolupament que Android ens proporciona amb unes llibreries SDK, un motor gràfic Unity addicional basat en la creació de videojocs per a poder desenvolupar les estructures 3D de l'aplicació, i SQL Server que ens permetrà crear una base de dades per a la gestió de tota la informació de les dades a la xarxa.

## 1.4 Eines utilitzades

- Entorn de desenvolupament integrat de codi obert multi plataforma Eclipse.
- Kit de desenvolupament de software o SDK d'Android.
- Java Development Kit (JDK).
- Netbook Samsung NP300 i semi torre AMD Athlon X2 Dual Core processor 4800+ 2.51Ghz.
- Samsung Galaxy SIII, Samsung Galaxy S i Samsung Galaxy Tab II.
- Servei de base de dades Oracle subministrat per la UAB.
- Servei de Web hosting subministrat per la UAB.
- Servei Google Cloud Messaging (GCM) gratuït de Google.
- Microsoft Project 2010 per a la creació de diagrames de seqüència i diagrames de Gantt.

## 1.5 Planificació temporal del treball

- **Fase 1 (Novembre - Desembre):** presa de decisions amb el tutor per a veure cap a quina direcció va el projecte. Consensuar objectius principals i assolir una planificació del projecte.
- **Fase 2 (Gener - Febrer):** Creació de la Base de Dades amb SQL Server i fer accessible aquesta informació des d'un dispositiu mòbil. Començar a desenvolupar la part teòrica de la simulació de tàctiques.
- **Fase 3 (Març - Juny):** Tenir enllestida la part teòrica i implementades les simulacions. Desenvolupar la interfície gràfica i les interaccions de l'aplicació.
- **Fase 4 (Maig - Juny):** Acabar de desenvolupar la interfície gràfica i preparar la memòria final.
- **Fase 5 (Juliol):** Finalització de la memòria.
- **Fase 6 (Juliol - Agost - Setembre):** tenir la presentació llesta per a l'exposició davant del tribunal.

Per a poder veure d'una manera més global la planificació del projecte tenim el diagrama de Gantt que es mostra a continuació:



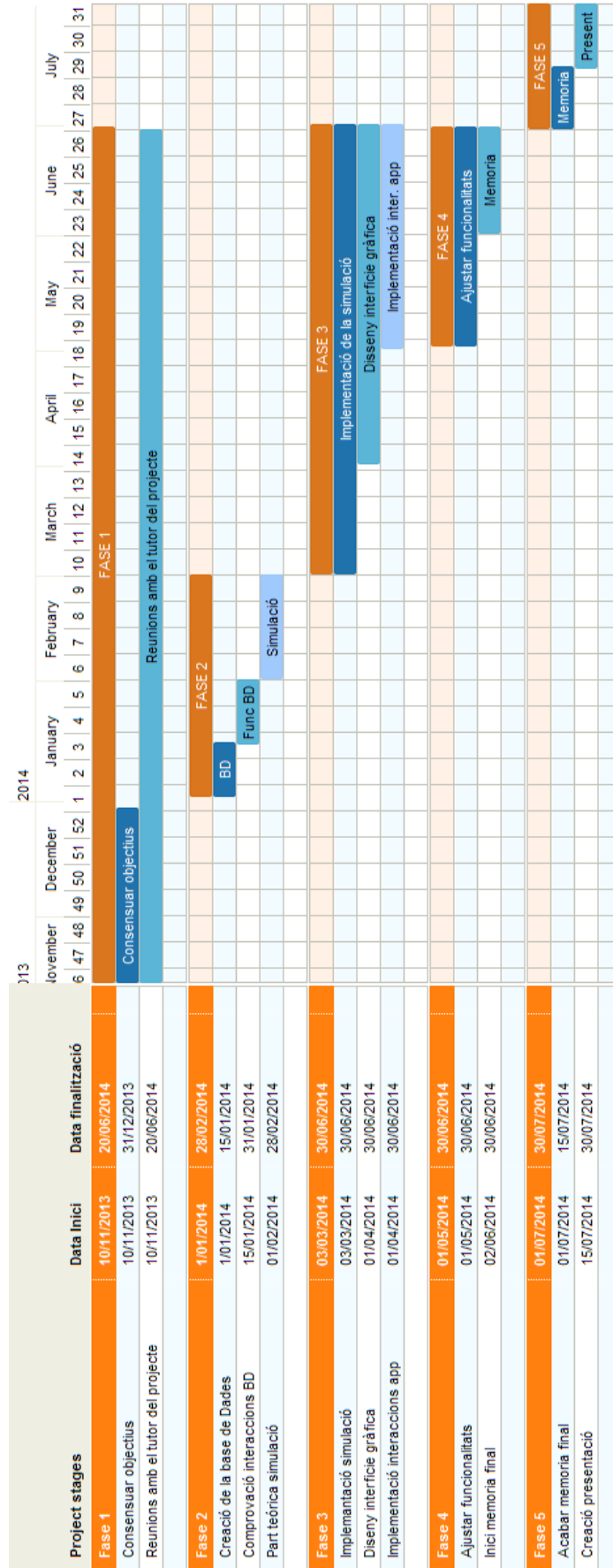


Figura 1: Diagrama de Gantt de la planificació de desenvolupament.

## 2. ESTAT DE L'ART

### 2.1 Dispositius mòbils

Definició: “*Els dispositius mòbils són aparells de mides reduïdes amb algunes capacitats de processament, amb connexió permanent o intermitent a una xarxa, amb memòria limitada i que són dissenyats específicament per a una funció concreta, però poden dur a terme altres funcions més generals.*” (Baz Alonso, Ferreira Artime, Álvarez Rodríguez, García Baniello, s.f., paràg. 2).

Tanmateix, personalment m'agrada més la definició que li dona W3C, que defineix el terme de dispositiu mòbil com: “*aparell portàtil, amb el qual es pot accedir a la web i dissenyat per a ser utilitzat en moviment*” (Nkeze; Pearce; Womer, 2007).

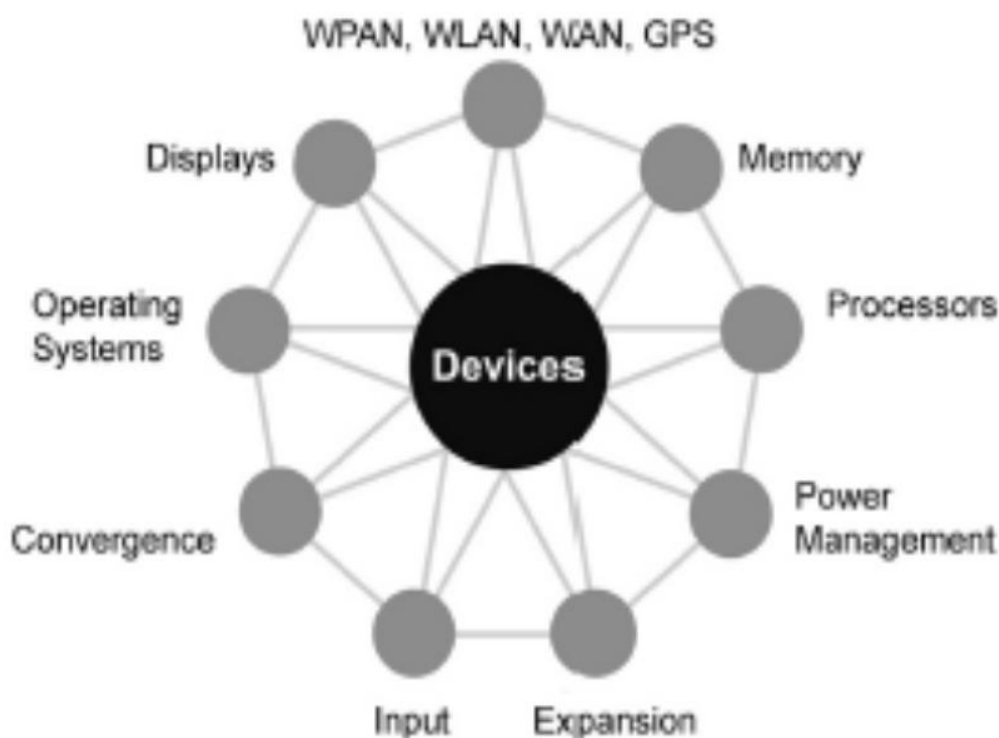
#### 2.1.1 Introducció

El concepte de “Dispositius mòbils” engloba una àmplia varietat de terminals amb funcionalitats molt diferents. Es tracta d'un mercat canviant, on cada any sorgeixen novetats i millores tecnològiques. Si bé els dispositius mòbils aporten mobilitat en la consulta de la Web, a més d'una millor connectivitat i accés a la informació, també presenten diverses limitacions derivades de les xarxes de comunicació, com la dependència d'una connexió telefònica o d'una xarxa wifi oberta, o marcades pels propis dispositius i el software.

Degut a les seves característiques i restriccions podem classificar els diferents dispositius mòbils en varies categories. **T38** i **DuPont Global Mobility Innovation Team** (any 2005) van proposar la següent classificació de dispositius mòbils:

- **Dispositiu Mòbil de Dades Limitades** (*Limited Data Mobile Device*): Referit a dispositius mòbils de pantalla petita de tipus text, on els serveis de dades solen estar limitats a l'enviament i recepció de missatges de text i connexions de tipus WAP.
- **Dispositiu Mòbil de Dades Bàsiques** (*Basic Data Mobile Device*): Referit a dispositius mòbils amb una pantalla de mida mitja i una navegació basada en icones. A més ofereixen enviament i recepció d'e-mails, missatges de text i navegador web entre d'altres funcionalitats. Aquest grup el formen terminals anomenats Smartphone.

- **Dispositiu Mòbil de Dades Millorats** (*Enhanced Data Mobile Device*): Referit a dispositius mòbils amb una pantalla de mida gran que ofereixen la mateixa funcionalitat que els Dispositius Mòbils de Dades Bàsiques a més d'aplicacions de Microsoft com poden ser Word i Excel entre d'altres. Entre els elements que poden arribar a ser un dispositiu mòbil (una PDA, un telèfon mòbil, un lector de llibres electrònics o un ordinador portàtil) destaca, sens dubte, el telèfon mòbil com a dispositiu més utilitzat entre tots.



**Figura 2:** Diagrama amb les característiques d'un dispositiu mòbil.

### 2.1.2 SmartPhone/Tablet

Un smartphone, o també anomenat telèfon mòbil intel·ligent (smartphone en anglès), és un terme comercial que s'utilitza per a denominar a un telèfon mòbil que ofereix més funcions que un telèfon mòbil comú. Per definició: *“Un smartphone és una evolució del telèfon mòbil tradicional que compta amb certes característiques i prestacions que l'acosten més a un ordinador que a un telèfon tradicional.”* (Aranaz Tudela, 2009).

Gairebé tots els telèfons intel·ligents són dispositius que suporten completament un client de correu electrònic amb funcionalitat completa d'un organitzador personal, que sembla ser una de les característiques indispensable trobada en tots els models existents des de 2007.

La característica més important (una d'elles) de gairebé tots els telèfons intel·ligents és que permeten la instal·lació de programes per incrementar el processament de dades i la connectivitat. Aquestes aplicacions poden ser desenvolupades pel fabricant del dispositiu, per l'operador o per un tercer, i és en aquesta última en la qual ens basem a l'hora de desenvolupar aquest projecte.

El terme "Intel·ligent" fa referència a qualsevol interfície, com ara un teclat QWERTY en miniatura, una pantalla tàctil (el més habitual, denominant-se en aquest cas "telèfon mòbil tàctil"), o simplement el sistema operatiu mòbil que posseeix. L'ús dels dispositius intel·ligents varia segons el sistema operatiu que integra, mitjançant una exclusiva disposició dels menús, tecles, dreceres, etc.

Les principals característiques d'aquests telèfons mòbils intel·ligents són:

- No només serveix com a dispositiu de comunicació (fer trucades) sinó que a més és un complet organitzador personal.
- Suporta correu electrònic.
- Compta de GPS.
- Permet instal·lació de programes de tercers.
- Utilitza qualsevol interfície per a l'ingrés de dades, com ara teclat QWERTY, pantalla tàctil.
- Permet connexió a internet.
- Posseeix agenda digital i administració de contactes.
- Permet llegir documents en diferents formats, entre ells els PDFs i arxius de Microsoft Office.
- Ha de comptar amb algun sistema operatiu.
- Es poden fer varies coses a la vegada (rebre trucades, revisar la teva agenda mentre sincronitzes el teu dispositiu amb altres, etc.).

Cada vegada més estan apareixent dispositius anomenats Tablets 3G, que adquireixen la funcionalitat de telèfon mòbil i només difereixen de la idea de smartphone en mida i potència, oferint les mateixes funcionalitats però augmentant les prestacions. Per tant, ho

englobarem tot dins de la mateixa categoria en quan al funcionament d'aplicacions.

La definició de tablet (de l'anglès tablet) es: *“Un dispositiu mòbil d'una mida més gran que un telèfon mòbil intel·ligent o una PDA, integrat en una pantalla tàctil (senzilla o multi tàctil) amb la qual s'interactua primàriament amb els dits o un llapis (passiu o actiu), sense necessitat de teclat físic ni ratolí.”*

## 2.2 Sistemes Operatius

Totes aquestes prestacions i característiques dels telèfons mòbils intel·ligents estarien desaprofitades sense un software que els hi tregui partit. Per això aquests dispositius duen un Sistema Operatiu (d'ara en endavant SO) que els permeten realitzar totes aquestes tasques de manera ràpida i senzilla.

Partint de la definició de SO: “Capa complexa entre el hardware i l'usuari, concebible també com una màquina virtual, que facilita a l'usuari o al programador les eines i interfícies adequades per a realitzar les tasques informàtiques, abstraient-lo dels complicats processos necessaris per a dur-les a terme.” (Baz Alonso, Ferreira Artime, Álvarez Rodríguez, García Baniello, s.f.). Per tant, un SO mòbil es un SO que controla un dispositiu mòbil. Però, aquests són molt més simples, n'hi ha més varietat respecte als altres sistemes operatius d'ordinador, i l'ús d'un SO o un altre determinarà la funcionalitat i la forma d'interactuar amb l'usuari.

La popularitat dels telèfons mòbils intel·ligents augmenta dia rere dia, i amb ella, la importància dels sistemes operatius amb els quals funcionen.

En el següent apartat realitzarem una breu descripció dels SO per a dispositius mòbils intel·ligents més importants en el mercat avui en dia.

### 2.2.1 Android

Android és un SO basat en la plataforma software de Linux. Aquest és lliure, gratuït i multi plataforma. Està dissenyat per al seu ús en telèfons mòbils intel·ligents encara que actualment també hi és disponible per a Tablets, rellotges intel·ligents, televisors i automòbils.

Mitjançant l'ús de biblioteques desenvolupades amb un llenguatge de programació Java, obtenim el control del dispositiu mòbil intel·ligent. Android és una plataforma de codi obert, això vol dir que qualsevol desenvolupador pot crear aplicacions escrites en un llenguatge de programació d'alt nivell i compilar-les després a codi natiu de ARM (API d'Android).

Android Inc. va començar a desenvolupar inicialment aquest SO, però Google Inc, que li va donar suport econòmic des de l'inici, va acabar comprant Android Inc. al 2005, i poc després es va unir a Open Handset Alliance: un consorci de companyies de hardware, software i

telecomunicacions per a avançar en els estàndards oberts dels dispositius mòbils. Google però, ha estat qui ha publicat la majoria del codi font d'Android sota llicència de Software Apache, una llicència de software lliure i de codi obert a qualsevol desenvolupador.

Com a principals característiques i especificacions podem destacar:

- **Disseny de dispositiu:** La plataforma és adaptable a qualsevol tipus de pantalla.
- **Maquina virtual Dalvik:** Base de crides de instàncies, molt semblant a Java i dissenyada específicament per a Android SO.
- **Entorn de desenvolupament:** Gratuït
- **Market:** Catàleg d'aplicacions per a la seva descàrrega en dispositius Android sense necessitat d'un ordinador.
- **Multitasca:** Les aplicacions en segon pla reben cicles de rellotge, a diferència d'altres sistemes.
- **Tethering (ancoratge de xarxa):** Ús del terminal com a punt d'accés sense fils des d'altres dispositius.

### 2.2.2 iOS

iOS és el sistema operatiu mòbil d'Apple. És una versió reduïda de Mac OS X. El iOS consta de 4 capes d'abstracció:

- Capa del nucli del sistema operatiu
- Capa de serveis principals
- Capa de medis de comunicació
- Capa de Cocoa Touch.

Actualment es troba en la seva 7a versió, que té com a principals característiques i especificacions:

- **Interfície d'usuari:** Interfície multi Touch. La interacció amb el SO es realitza mitjançant gestos amb el dit com lliscar, prémer i pessigar.
- **Multitasca:** A partir de iOS 4 però, no consisteix en una veritable multitasca, atès que les aplicacions independents del SO, queden

congelades en segon pla no reben ni un cicle de rellotge del processador.

- **Disseny sobre segur:** Totes les aplicacions s'executen en un entorn segur, així que cap pàgina web ni cap aplicació poden accedir a les dades de les altres aplicacions instal·lades en el dispositiu. Com iOS és compatible amb la comunicació en xarxa xifrada, la informació confidencial sempre està protegida. Per salvaguardar la privacitat de l'usuari, les aplicacions que sol·liciten informació sobre la seva ubicació hauran de comptar amb el consentiment previ de l'usuari.
- **Accessibilitat integrada:** iOS incorpora nombroses prestacions de accessibilitat per a que persones amb discapacitat puguin gaudir de tot el que ofereix el dispositiu mòbil. Per exemple, la tecnologia integrada de lectura de pantalla VoiceOver permet als invidents i a persones de visibilitat reduïda escoltar una descripció dels elements de la pantalla.
- **Seguretat:** Abans de la sortida de iOS 7 al mercat, existia un gran índex de robatoris de diversos model de dispositiu, fet que el govern nord-americà va sol·licitar a Apple dissenyar un sistema de seguretat infal·libre que inutilitzava els dispositius mòbils en cas de robatori. A més és possible consultar la ubicació del dispositiu via GPS i mostrar els missatges de pantalla. Aquesta funcionalitat el converteix en el SO més segur del mercat.

### 2.2.3 Windows Phone

Windows Phone es un SO mòbil dissenyat per telèfons mòbils intel·ligents i d'altres dispositius mòbils. Forma part dels SO que compten amb una interfície d'usuari natural. Basat en el nucli de Windows CE, està dissenyat per ser estèticament similar a les versions d'escriptori de Windows.

Un any després de la seva primera versió, llançava al mercat la seva primera actualització, Windows Phone 7, i posteriorment, al 2012, Windows Phone 8, i més tard la versió 8.1 que té com a principals atractius:

- **Interfície Modern UI:** On la pantalla d'inici es compon de LiveTiles, mosaics dinàmics que són enllaços a les aplicacions o objectes individuals. La pantalla d'inici i els mosaics dinàmics es poden personalitzar prement i lliscant els mosaics a la posició desitjada.



- **Estalvi energètic:** La interfície d'usuari per defecte té un estil visual negre, que prolonga la bateria en pantalles LED. L'usuari pot escollir el tema de visualització de la interfície al seu gust.
- **Hubs:** Windows Phone no és un sistema centralitzat únicament en aplicacions sinó que s'organitza en un nou concepte denominat hub. Els hubs es classifiquen en accions i agrupen aplicacions que es corresponen amb una activitat determinada. Així en Windows Phone hi ha hubs de Contactes, Office i Jocs. Les aplicacions de tercers es poden integrar en el hub que correspongui millorant la seva funcionalitat.
- **Office:** Windows Phone conté un hub d'Office, on es pot accedir a Word, Excel, OneNote i PowerPoint. Aquestes aplicacions ens permeten veure, editar i compartir arxius d'aquests serveis a través d'OneDrive i SharePoint. El hub Office permet realitzar comentaris i correccions sobre documents i treballar de manera col·laborativa amb aquests usuaris.
- **MarketPlace:** Tot i ser una aplicació de compra d'aplicacions, també es poden provar les aplicacions abans de comprar-les, cosa que no tenen les altres plataformes mòbils.

## 2.2.4 BlackBerry OS

El BlackBerry SO és un SO mòbil desenvolupat per Research Inmotion (RIM) per a dispositius mòbils BlackBerry. Aquest SO permet a l'usuari gaudir de la multi tasca i a més té suport per a diferents mètodes d'entrada adoptats per RIM per al seu ús en dispositius mòbils, particularment la trackwheel, trackball, touchpad i pantalles tàctils.

El seu desenvolupament es remunta a l'aparició dels primers handheld (dispositius mòbils) el 1999. Aquests dispositius permeten l'accés a correu electrònic, navegador web i sincronització amb programes com Microsoft Exchange o Lotus Notes, a part de tenir la funcionalitat d'un telèfon mòbil.

Aquest SO està clarament orientat al seu ús professional atès que disposa de correu electrònic i agenda. Com en el SO Symbian (desenvolupat per Nokia), els desenvolupadors poden crear aplicacions per a BlackBerry, però en el cas de voler tenir accés a certes funcionalitats restringides, necessiten ser firmats digitalment per a poder associar-los a una compta de desenvolupador de RIM.

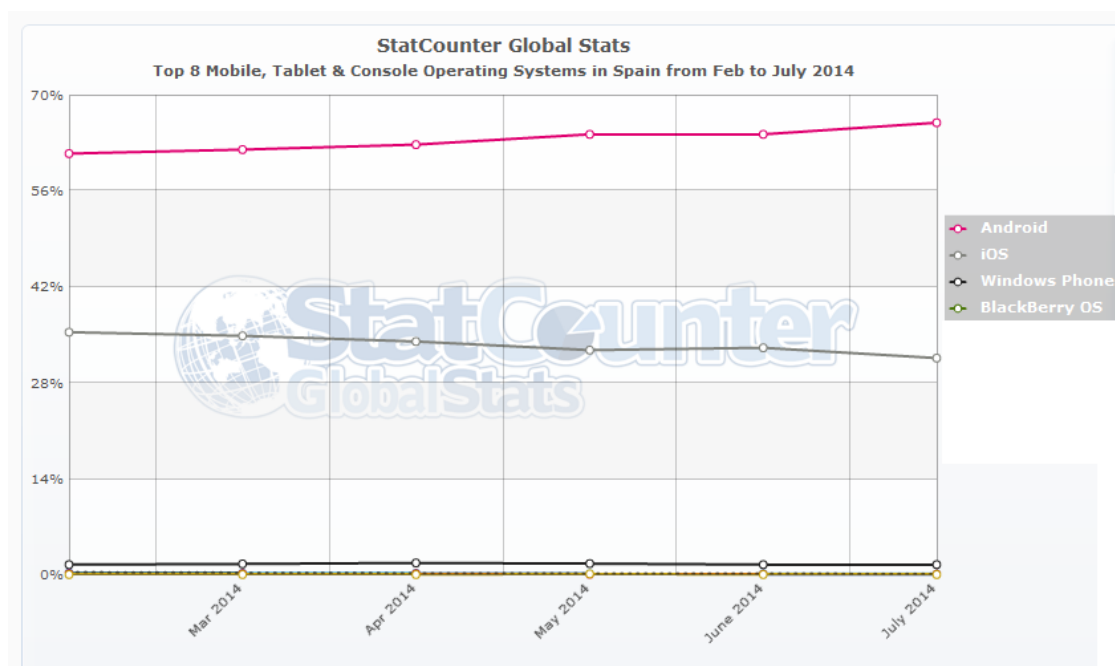
## 2.3 Justificació de l'elecció

Per a l'elecció de la plataforma que més s'ajusta a les meves necessitats i interessos, he tingut en compte diferents factors a valorar per a dur a terme el projecte de forma satisfactòria, que estan llistats a continuació:

- **Quota de mercat:** Quantitat de dispositius que integren de sèrie la plataforma indicada.
- **Preu per a desenvolupador:** Preu de la llicència per a desenvolupar les aplicacions en cada plataforma.
- **Nombre d'aplicacions:** Quantitat d'aplicacions disponibles actualment en el mercat de cada plataforma.
- **Expectatives:** Tendència dels nous dispositius mòbils a incorporar la plataforma determinada.

### Quota de mercat:

Actualment, segons els estudis realitzats per StatCounter, la quota de mercat dels sistemes operatius mòbils des de Febrer fins al juliol de 2014 es divideix de la següent manera:



**Figura 3:** Diagrama quota de mercat dispositius mòbils segons sistema operatiu.

En aquest gràfic podem veure que la plataforma Android és implementada en la gran majoria de dispositius mòbils a Espanya.

### Preu per a desenvolupador:

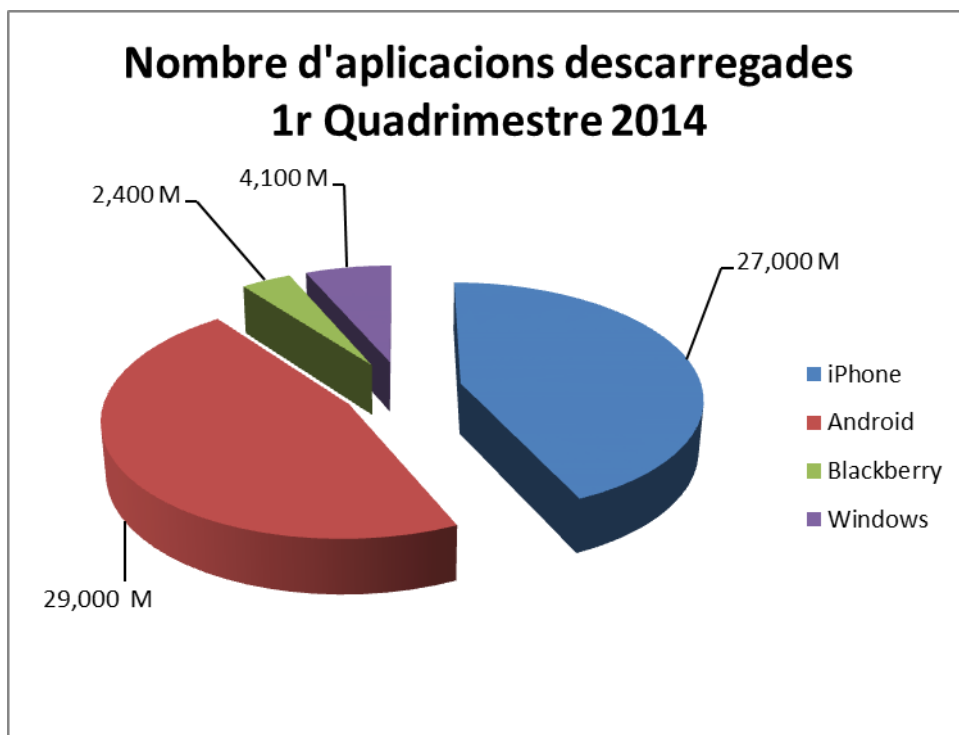
Un dels factors més importants per a decidir la plataforma on es desenvoluparà l'aplicació és el preu que han de pagar tots els desenvolupadors que vulguin crear aplicacions en alguna de les plataformes més importants actualment.

Sistema Operatiu	Preu de llicència/any
Android	25 \$
iOS	99 \$ + 299 \$ Apple Store
Windows Phone	99 \$
BlackBerry OS	Gratuït

En el llistat de les principals plataformes veiem que el preu de llicència de la plataforma Android és molt assequible a nivell personal i a més no s'hauria de pagar cap llicència per a desenvolupar l'aplicació, només s'hauria de pagar per pujar-la al mercat.

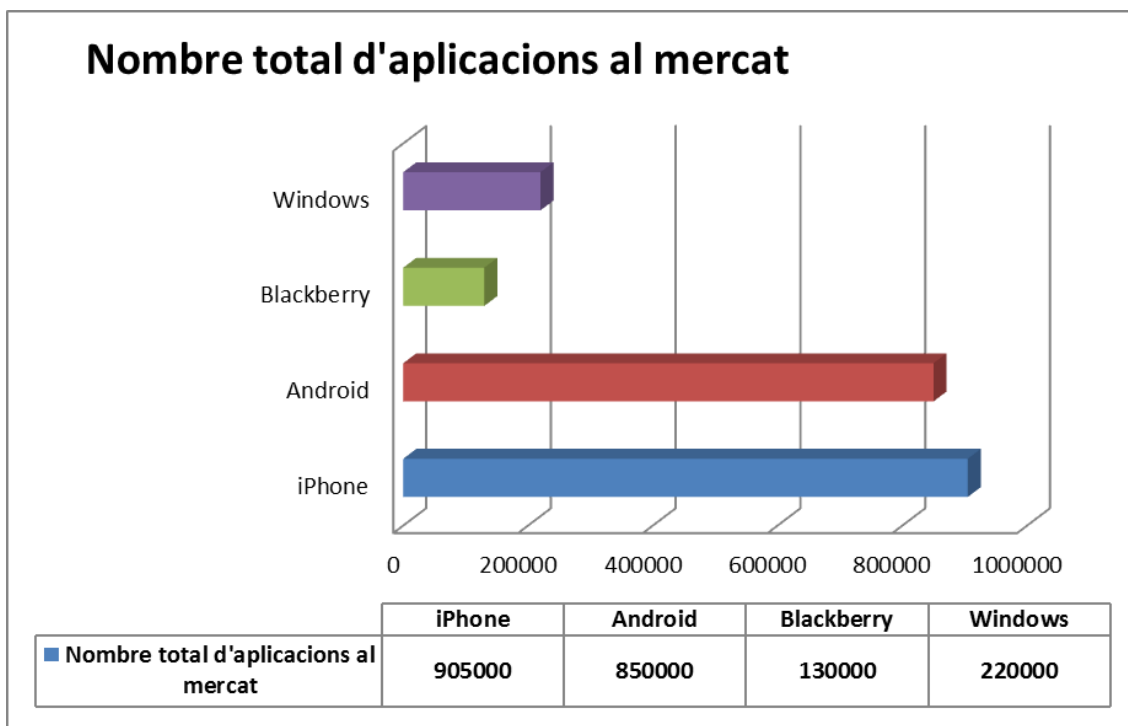
### Nombre d'aplicacions:

Per saber el nombre d'aplicacions disponibles actualment en el mercat he consultat Statistic Brain que ens ofereix les dades de Juliol de 2014. En aquest primer gràfic veiem la quantitat de descàrregues d'aplicacions que s'han generat en el transcurs del primer quadrimestre de 2014 en termes globals.



**Figura 4:** Diagrama sobre el nombre d'aplicacions descarregades durant el 1r Quadrimestre 2014 segons sistema operatiu.

Observem que casi la totalitat de descàrregues es duen a terme a través de les plataformes Android i iOS, una dada lògica ja que la majoria de dispositius al mercat formen part d'aquests dos sistemes operatius. En el següent gràfic hi podem veure el global d'aplicacions disponibles al mercat de cada plataforma.

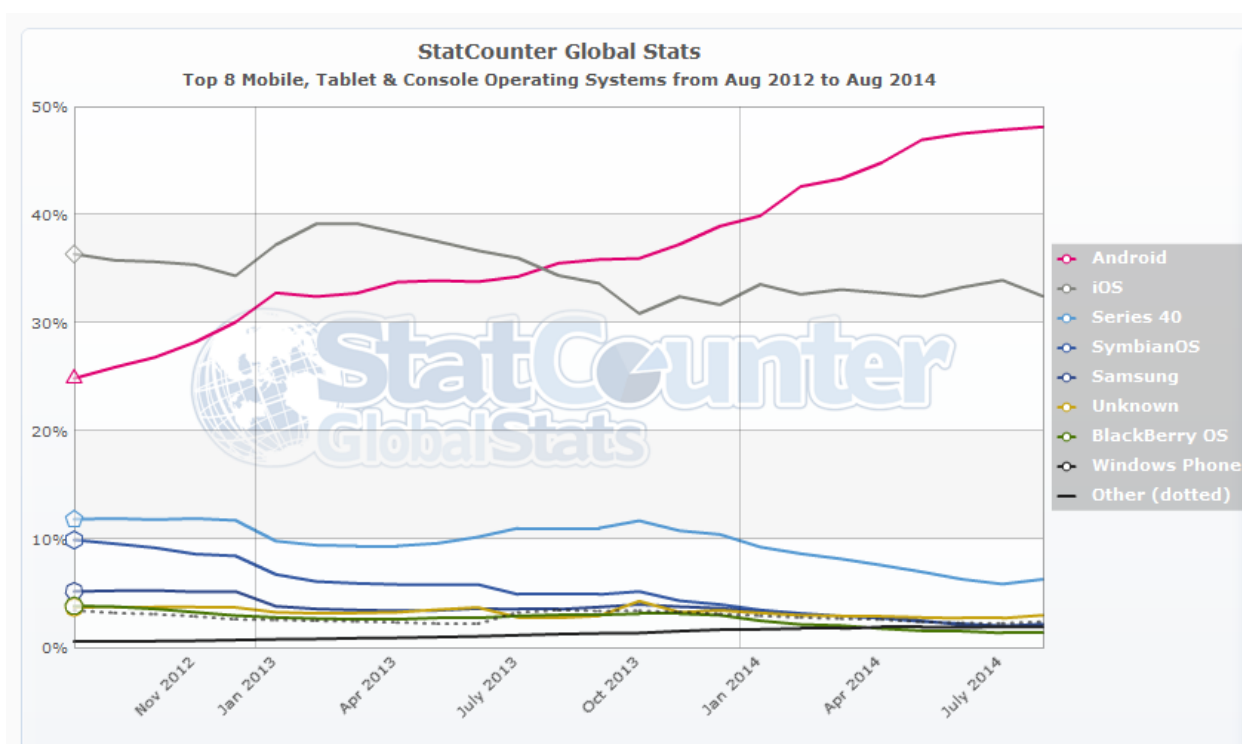


**Figura 5:** Diagrama del nombre total d'aplicacions al mercat segons sistema operatiu.

On podem veure que Android i iOS segueixen dominant en quan a desenvolupament d'aplicacions, probablement per la quantitat de dispositius mòbils que integren les dues plataformes.

### Expectatives:

Una raó de pes alhora d'escollir la millor plataforma per desenvolupar, és la tendència que tindrà el mercat. No seria igual de productiu fer una aplicació per una plataforma que té tendència a desaparèixer, que fer-la per a una plataforma que té expectatives de mantenir-se en el mercat durant força temps. Per això, és interessant examinar el següent gràfic extret de StatCounter, on s'hi pot observar la tendència en els últims 2 anys de la quantitat de dispositius mòbils que integren cada plataforma.



**Figura 6:** Diagrama d'evolució de la venda de dispositius d'entre 2012 i 2014.

Veiem doncs que la plataforma Android és la que té més tendència a seguir augmentant en quant a quantitat de dispositius.

En conclusió, la millor opció per desenvolupar aplicacions mòbils per a dispositius amb les diferents plataformes és Android o iOS. D'aquesta manera justifico la implementació d'una aplicació per a un sistema operatiu Android d'acord amb els següents punts:

- Sistema operatiu basat en codi obert, motiu pel qual el suport és global, amb una pila de programadors a tot el món disposats a corregir errors en moments en què no es troba una solució.
- Les aplicacions es desenvolupen en llenguatge de programació Java, un llenguatge madur, àmpliament adoptat i usat en moltes de les matèries i pràctiques dutes a terme durant el curs.
- Té una SDK multiplataforma que funciona en Windows, Linux i Mac, mentre que per crear aplicacions iOS ens faria falta un ordinador de la marca Apple.
- Multitud de dispositius mòbils de diferents marques que integren aquest SO. En canvi en iOS només es podria utilitzar els models de la marca Apple.
- La majoria de dispositius Android suporten més capacitat de memòria via targetes SD, en canvi els Apple no.

## 3. ENTORN DE TREBALL

### 3.1 Android

Com ja hem dit abans, per a treballar amb la plataforma Android, Google ens proporciona diverses eines de treball per a desenvolupar aplicacions.

Des de la pàgina d'Android Developers se'ns proporciona un paquet software que ja conté el programa de desenvolupament Eclipse juntament amb el SDK d'Android per a poder utilitzar les seves funcions i poder emular un simulador des de el mateix programa. Això ens serveix de gran ajuda a l'hora de provar l'aplicació si no disposem d'un telèfon mòbil o Tablet amb el mateix sistema operatiu. Aquest paquet té el nom de ADT, que vol dir Android Development Tool (Eina de desenvolupament d'Android).



**Figura 7:** Caràtula d'inici d'Eclipse.

En primer lloc, com que el SDK d'Android es compon de funcions escrites en Java, serà necessari instal·lar l'última versió del JDK (Java Development Kit) a l'ordinador on desenvoluparem l'aplicació. El podem trobar a la pàgina oficial d'Oracle.

Un cop descarregat el paquet software i instal·lat el JDK, haurem d'actualitzar els mòduls del SDK d'Android que necessitem segons la versió de SO que utilitzem alhora de desenvolupar el projecte.

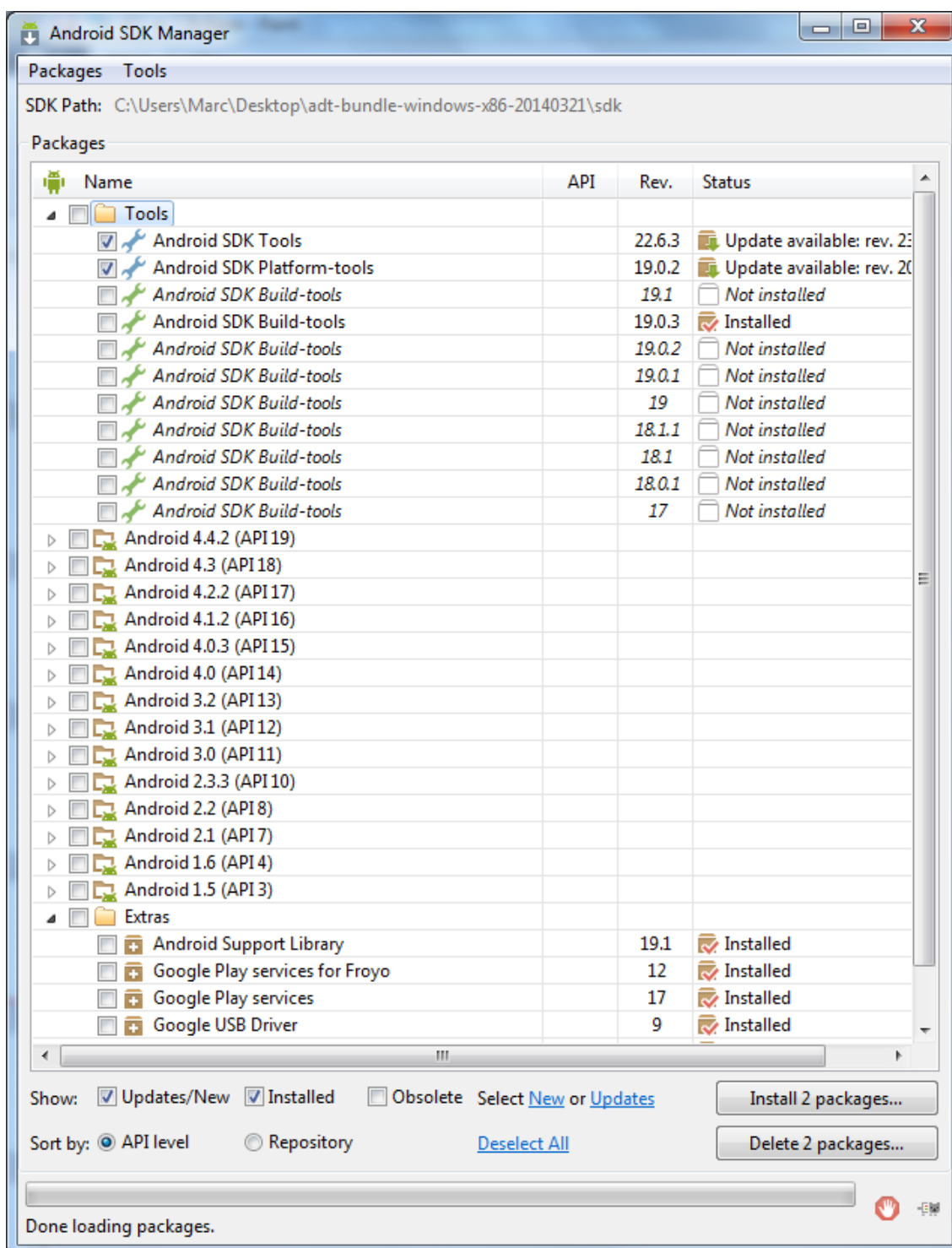


Figura 8: Android SDK Manager.

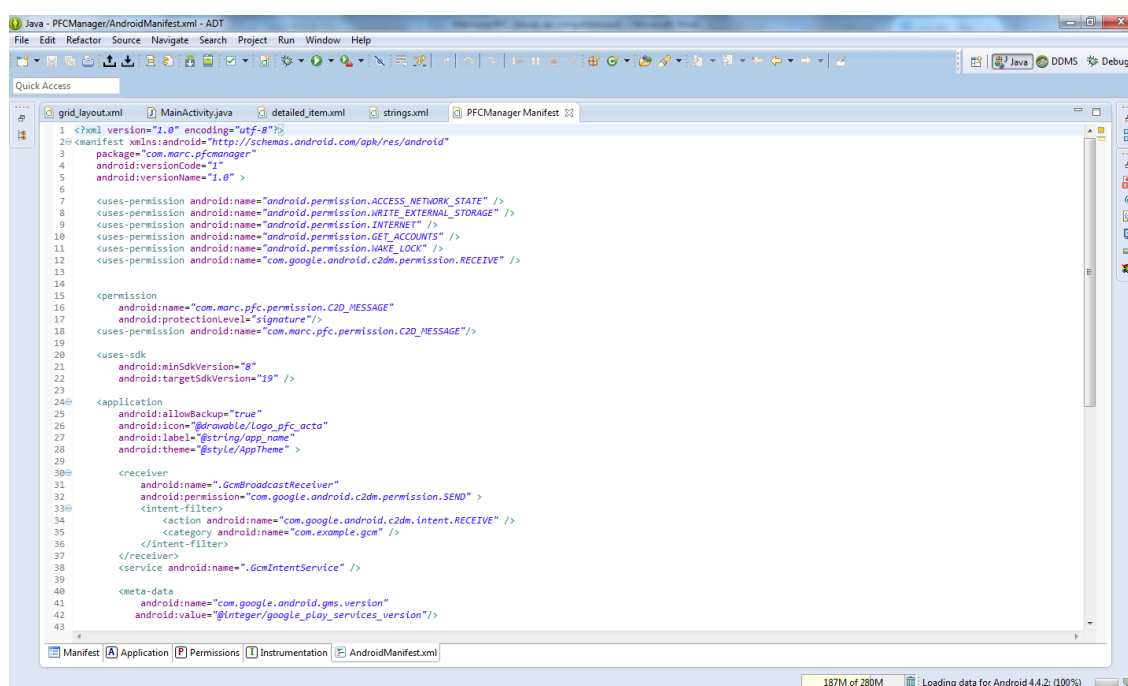
Cada vegada que es comença a desenvolupar una nova aplicació Android sorgeix la pregunta: Quin es la mínima API (versió d'Android) que haig d'utilitzar? Si escollim una API molt baix pensant en poder abastar la majoria de dispositius del mercat, llavors perdrem funcions noves i importants que utilitzen les versions més actuals, i si utilitzem una versió molt nova/SDK molt alt, podríem quedar atrapats en un petit sector del



mercat, ja que no tothom té dispositius mòbils d'altres prestacions amb versions d'Android actuals.

En primer lloc, haurem de planificar i pensar molt bé l'aplicació molt abans de començar a escriure el codi. És possible que sigui necessari utilitzar per força major una funcionalitat que està en un SDK d'una versió nova. En aquests casos haurem de veure si realment és vital per al projecte.

Seguint amb el software de desenvolupament Eclipse, un cop instal·lat el JDK i configurat el SDK, veiem que la base és una plataforma de client enriquit (Rich Client Platform, RCP). La definició que dona el projecte Eclipse sobre el seu software és: *“una espècie d'eina universal – un IDE obert i extensible per a tot i per a res en particular”*



```

1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     package="com.marc.pfcmanager"
4     android:versionCode="1"
5     android:versionName="1.0" >
6
7     <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
8     <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
9     <uses-permission android:name="android.permission.INTERNET" />
10    <uses-permission android:name="android.permission.GET_ACCOUNTS" />
11    <uses-permission android:name="android.permission.WAKE_LOCK" />
12    <uses-permission android:name="com.google.android.c2dm.permission.RECEIVE" />
13
14
15    <permission
16        android:name="com.marc.pfc.permission.C2D_MESSAGE"
17        android:protectionLevel="signature"/>
18    <uses-permission android:name="com.marc.pfc.permission.C2D_MESSAGE"/>
19
20    <uses-sdk
21        android:minSdkVersion="8"
22        android:targetSdkVersion="19" />
23
24    <application
25        android:allowBackup="true"
26        android:icon="@drawable/logo_pfc_acto"
27        android:label="@string/app_name"
28        android:theme="@style/AppTheme" >
29
30        <receiver
31            android:name=".GcmBroadcastReceiver"
32            android:permission="com.google.android.c2dm.permission.SEND" >
33            <intent-filter>
34                <action android:name="com.google.android.c2dm.intent.RECEIVE" />
35                <category android:name="com.example.gcm" />
36            </intent-filter>
37        </receiver>
38        <service android:name=".GcmIntentService" />
39
40        <meta-data
41            android:name="com.google.android.gms.version"
42            android:value="@integer/google_play_services_version"/>
43    </application>

```

Figura 9: Entorn de desenvolupament Eclipse.

L'entorn de desenvolupament integrat d'Eclipse (IDE) utilitza mòduls per a proporcionar tota la seva funcionalitat a la plataforma de client enriquit, a diferència d'altres entorns monolítics en què les funcionalitats estan totes incloses, les necessiti o no l'usuari. Aquest mecanisme de mòduls és una plataforma lleugera per a components software.

Com hem vist abans, el paquet software ADT proporcionat per Android Developers que integra el SDK d'Android, és un d'aquests mòduls de funcionalitat, que utilitza Eclipse per a poder integrar-se en els dispositius mòbils.

## 3.2 Unity

Com que abans no hem introduït Unity, ja que no és un sistema operatiu, aprofitarem ara per veure què és en essència.

Unity és un motor gràfic 3D, creat per Unity Technologies, per a gran quantitat de sistemes operatius que ve encapsulat com a eina de desenvolupament per a crear videojocs, aplicacions interactives, visualitzacions i animacions de 3D i temps real. Està disponible en la majoria de sistemes operatius del mercat com poden ser Windows, OS X, Linux, Android, PlayStation, Xbox, etc. El motor també pot publicar jocs basats en web, utilitzant el plugin Unity Web Player.

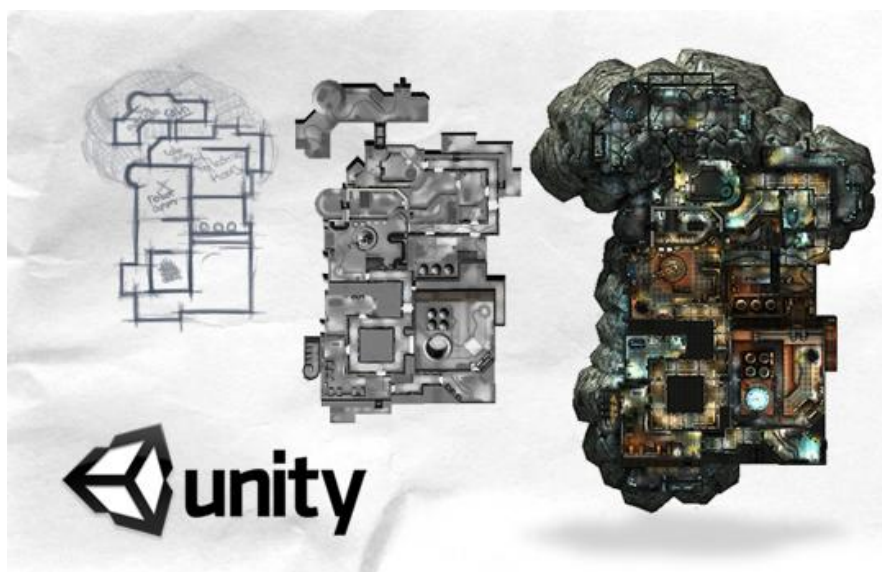


Figura 10: Caràtula d'inici de l'editor de Unity.

L'editor de Unity és l'element principal de la línia de producció, oferint un editor visual complet per a crear videojocs. El contingut del videojoc es construeix des de l'editor i el gameplay es programa utilitzant un llenguatge de scripts. Això vol dir que els desenvolupadors no necessiten ser uns experts en C++ per a crear els videojocs amb Unity, ja que les mecàniques de joc són compilades utilitzant una versió de JavaScript, C# o Boo, un dialecte de Python.

Els jocs creats a Unity s'estructuren en escenes, com ara el motor Gamebryo. A Unity, una escena pot ser qualsevol part del joc: des del menú d'inici, fins a un nivell o àrea del teu joc; l'elecció és de cada u, ja que una escena és un quadre en blanc sobre el qual dibuixar cada part del joc utilitzant les eines de Unity. El motor també incorpora un editor de terrenys, des d'on es pot crear un terreny (com un full en blanc), sobre el qual els artistes podran esculpir la geometria del terreny utilitzant eines

visuals, pintar o texturitzar, cobrir d'herba o col·locar arbres i d'altres elements de terreny importats des de aplicacions de modelatge en 3D com són 3D Studio Max, Maya, o Blender.

Amb això, donem per conclosa la introducció a Unity, i passarem a veure el seu entorn de treball.

Per a treballar amb Unity el més senzill és descarregar el paquet que Unity3D ens proporciona des de la seva pàgina web. Dins hi trobem tant l'editor de scripts, que es diu MonoDevelop, com l'editor de Unity i el motor gràfic 3D.

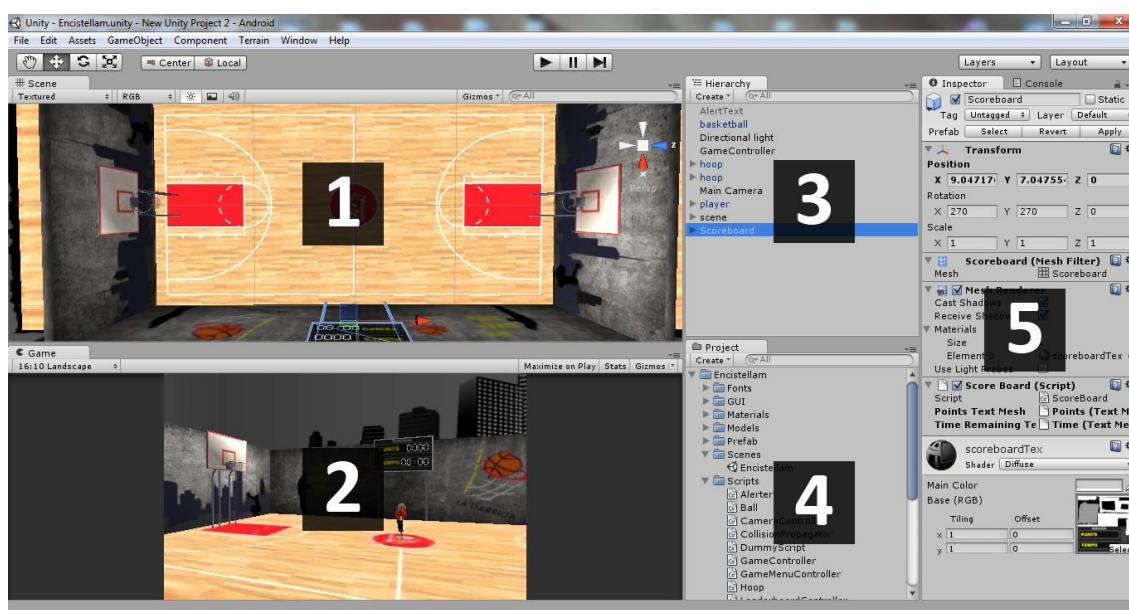


Figura 11: Entorn de desenvolupament de l'editor de Unity.

Un cop tot instal·lat, veurem la interfície de Unity. Hi existeixen principalment 5 àrees:

- **Vista de l'escena:** L'escena es l'àrea de construcció de Unity, on construïm visualment cada escena del nostre videojoc. Aquí podem moure, posicionar, rotar i transformar totes les figures/objectes que hi ha en la nostra escena per tal que tingui la distribució desitjada.
- **Vista de joc:** En aquesta vista obtindrem una previsualització del nostre joc. En qualsevol moment podem reproduir el nostre joc i jugar-lo en aquesta vista. Podríem dir que és una compilació en temps real i podem jugar els canvis fets prèviament sense haver d'exportar-lo en un altre format, com passa en Android.

- **Vista de projecte:** Aquesta és la llibreria d'Assets per al nostre projecte, similar a la llibreria Flash. Es poden importar objectes 3D de diferents aplicacions a la llibreria, així com textures i crear altres objectes com scripts o prefabs que s'emmagatzemen aquí per poder ser utilitzats pel nostre videojoc.

Ja que el videojoc normal contindrà diverses escenes i una gran quantitat d'assets, és una bona idea estructurar la llibreria en diferents carpetes que faran que els nostres assets es trobin organitzats i sigui més fàcil treballa-hi.

- **Vista de Jerarquia:** La vista de jerarquia conté tots els objectes que es troben en l'escena actual, és a dir, és un organitzador dels elements amb els quals estem treballant actualment en l'escena en pantalla.
- **Vista d'inspector:** La vista d'inspector serveix per a diferents accions. Si seleccionem objectes, aleshores ens mostrarà les propietats d'aquest objecte i podem personalitzar varies de les seves característiques. També conté la configuració per a certes eines com l'eina de terrenys si tenim el terreny seleccionat.

Un cop vist l'editor Unity veiem ara l'editor de scripts. Aquests scripts són escrits en MonoDevelop, que utilitza llenguatge C-Sharp.

Quan creem un script, podem saltar de l'editor de scripts desde Unity, guardar el script i utilitzar-lo en el joc. Inclou una manera dinàmica de poder assignar els scripts a diferents objectes del joc dinàmicament assignant a l'objecte un comportament escrit amb MonoDevelop.

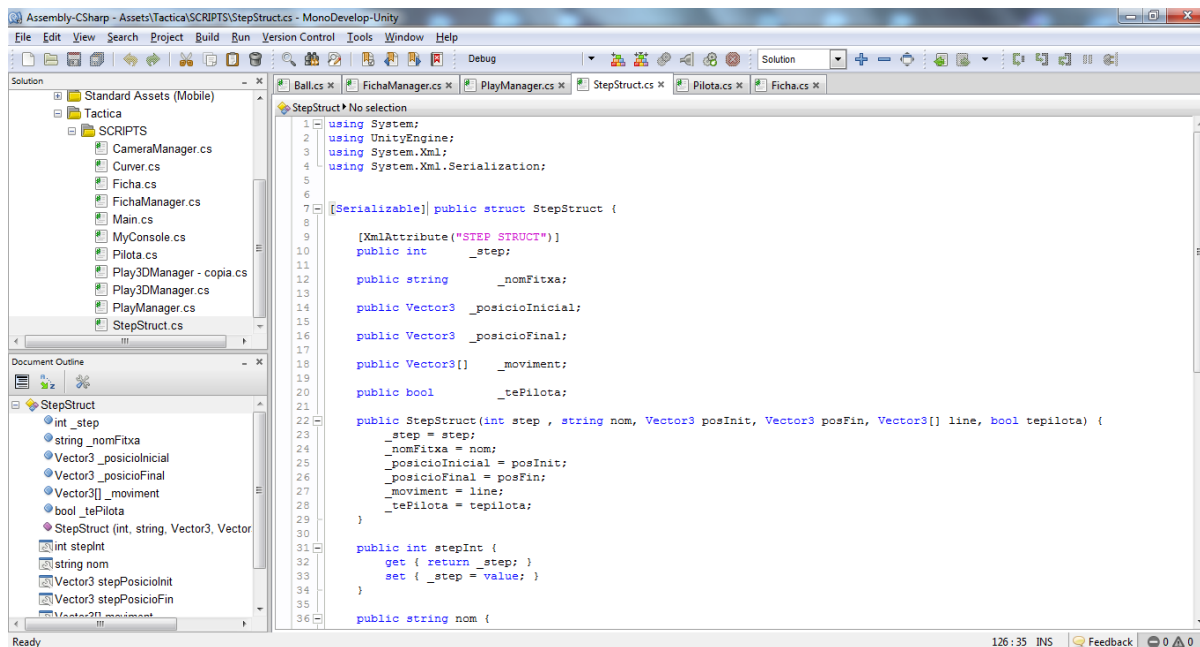


Figura 12: Entorn de desenvolupament MonoDevelop.

### 3.3 SQL Developer

Veiem doncs l'Oracle SQL Developer, que és una eina gràfica gratuïta distribuïda per Oracle per a que no sigui necessari utilitzar eines de tercers per a desenvolupar, o simplement per executar consultes o scripts SQL, tant DML com DDL, sobre bases de dades Oracle.



**Figura 13:** Caràtula d'inici de Oracle SQL Developer.

L'aparença i funcionalitat és similar a la d'altres eines d'aquest tipus, motiu pel qual és una bona opció si no tenim especial predilecció per altres eines de desenvolupament.

A més en les últimes versions ha incorporat millores com ara permetre la connexió amb bases de dades no Oracle, com SQLServer, MySQL o Access. La connexió amb MySQL o SQLServer es realitza a través de JDBC, i de manera bastant senzilla. Un cop establerta la connexió es poden explorar els objectes de la base de dades, com si es tractés d'una d'Oracle, i executar sobre elles sentències SQL.

El fet que durant els cursos de la carrera s'hagi introduït aquesta eina facilita bastant la comprensió del què s'està fent en cada moment, i per aquest motiu és la eina que he decidit utilitzar.

En resum, és una eina molt interessant alhora de gestionar informació de la base de dades, tan crear taules i relacions, introduir dades a les diferents taules o simplement fer consultes a la base de dades per a comprovar el seu correcte funcionament.

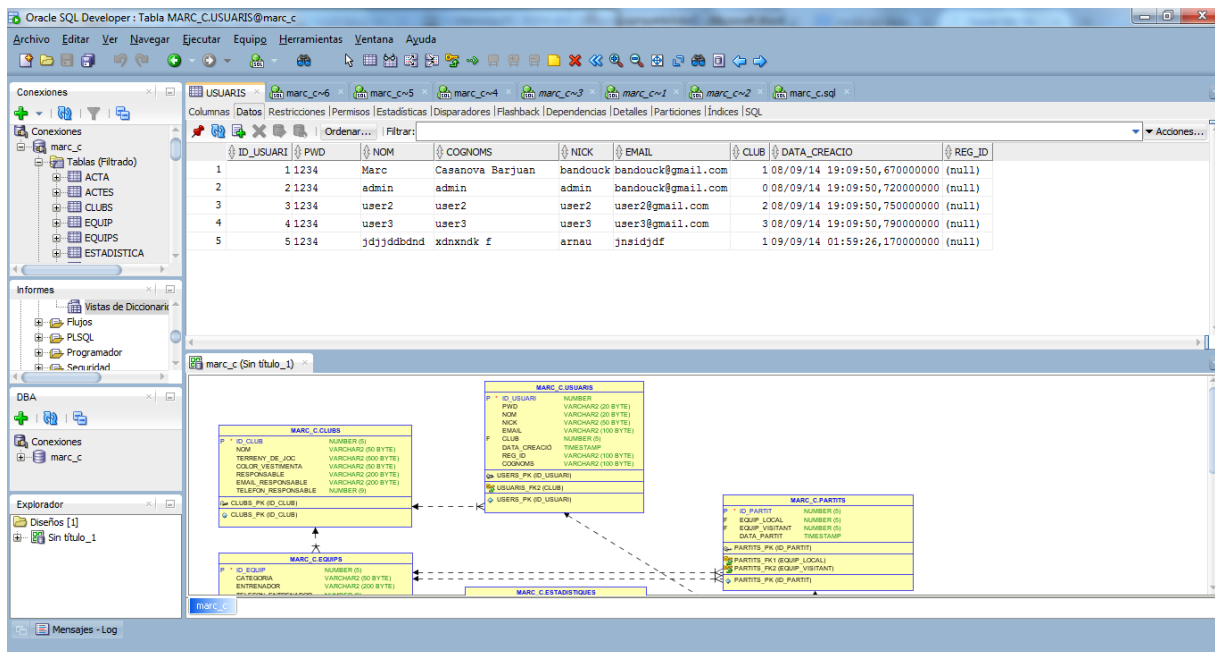


Figura 14: Entorn de desenvolupament d'Oracle SQL Developer

### 3.4 Anàlisi de requeriments

Els requeriments són cada un dels models del procés de desenvolupament del software proposat. Per tant, la comprensió dels propòsits i funcions del sistema comencen amb un atent examen dels requeriments.

Quan un client sol·licita que es desenvolupi un sistema, ha de tenir algunes nocions del que s'ha de fer. Per aquesta raó, cada sistema basat en software ha de tenir un propòsit, normalment expressat amb el que el programa ha de fer.

Un requeriment *“és una característica del sistema o una descripció d'alguna cosa que el sistema és capaç de fer amb l'objectiu de satisfer el propòsit del sistema”*.

En aquest apartat veurem les necessitats que tindrien els usuaris de l'aplicació tenint en compte la seva funció dins del club de bàsquet. Amb això vull dir que els requeriments que té un entrenador són diferents dels que té un delegat o una persona que no està físicament en el transcurs d'un partit o un entrenament.

Veiem els diferents usuaris que poden haver-hi i les diferents necessitats que poden tenir alhora d'utilitzar l'aplicació.

- **Entrenador:** És la persona responsable de dirigir un equip de bàsquet. Les seves necessitats o requeriments de l'aplicació són:
  - Pissarra: La pissarra és una eina avui en dia indispensable per a l'entrenador, ja que en ella pot dibuixar la tàctica o jugada a dur a terme en un punt determinat del joc. Normalment s'utilitza per a recordar als jugadors una jugada o fer una variant d'aquesta en un moment determinat.
  - Estadística: L'estadística és una manera probabilística de comprovar l'efectivitat, la participació i les característiques de cada jugador o global de l'equip en un partit o en el còmput general de la temporada.
  - Tàctica: La tàctica com a editor i reproductor. Segons l'entrenador les tàctiques a utilitzar varien, és a dir, cada entrenador és propens a utilitzar un tipus de jugades, ja que la visió del joc pot variar segons el partit, els jugadors i en definitiva, de l'equip rival. Per tant, seria necessària una interfície on es puguin crear les jugades en moviment i



emmagatzemar-les per a una posterior reproducció. En el moment indicat facilita el mecanisme de memorització de jugades tant en un entrenament com aplicant-ho en el partit.

- Simulador de defenses: La simulació d'un joc permet veure quina serà la tendència d'aquest joc aplicant certs paràmetres. Aquí l'entrenador pot simular segons la defensa que li estant fent al seu equip quina de les seves jugades d'atac pot arribar a ser més ofensiva. Això facilita la visió de l'entrenador pel que fa a les jugades que podria utilitzar durant un partit.
- **Delegat**: El delegat és l'encarregat de gestionar la part no física d'un equip de bàsquet durant un partit, com ara la coordinació de l'ordre en el terreny de joc.
  - Acta digital: L'acta és el full de paper on s'enregistren les anotacions, faltes, i resultats parcials del marcador durant el partit, i és el que la Federació té en compte en quan a la disputa d'un partit entre dos equips. Una acta digital (digital perquè s'enregistra des de un dispositiu electrònic) podríem dir que seria el mateix, però afegint-hi altres aspectes del joc determinants com els taps, les assistències, el percentatge de llançament, etc. determinants en la gestió d'un equip.
- **Jugador**: El jugador és l'encarregat d'executar els moviments establerts en la tàctica en la disputa dels partits de bàsquet.
  - Estadística: L'estadística és una manera probabilística de comprovar l'efectivitat, la participació i les característiques de cada jugador o global de l'equip en un partit o en el còmput general de la temporada.
  - Tàctica: La tàctica es pot editar i reproduir. La tàctica a utilitzar varia segons el jugador i segons les ordres de l'entrenador, a més cada entrenador és propens a utilitzar un tipus de jugades o unes altres. La tàctica també varia segons el partit, els jugadors o l'equip rival. Per tant, seria necessària una interfície que pugui reproduir jugades guardades per l'entrenador en qualsevol moment, així aconseguim un aprenentatge més dinàmic.
- **Soci**: Un soci del club de bàsquet és una persona que està lligada al club per diferents motius, com ser aficionat al club, ser família d'algun jugador o directament per ser jugador lesionat. Depenent de cada cas,

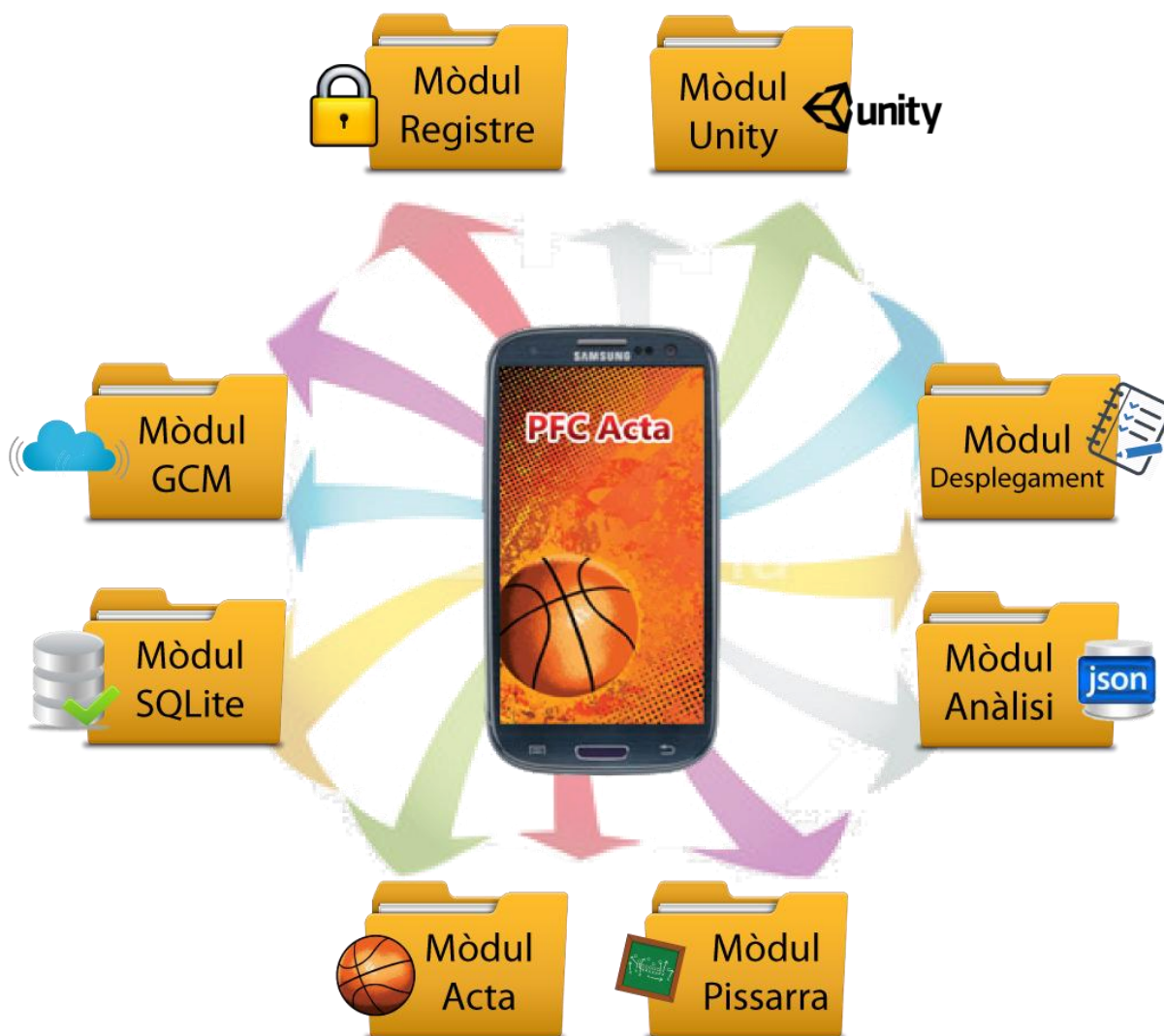
però englobant-ho dins del mateix conjunt d'usuaris els seus requeriments podrien ser:

- Acta digital: Aquesta acta digital també la podrien fer altres usuaris del club des de la mateixa graderia, ja que no varia massa la visió del joc del delegat o d'espectador. Amb això podríem tenir dues actes diferents del mateix partit duta a terme per dues persones diferents, tenint en compte el marge d'error de cada persona. Amb això aconseguim que si per casualitat un dia el delegat no pot assistir al partit, una persona aficionada podria dur a terme aquesta acta.
- Consulta acta digital: La consulta de l'acta digital mentre aquesta s'està duent a terme seria una de les practiques més innovadores d'aquest projecte. Mentre el delegat o aficionat està anotant, la mateixa aplicació enviaria un missatge a tots els dispositius que tinguessin instal·lada l'aplicació i formessin part del mateix club, un avís indicant que s'està jugant aquest partit juntament amb totes les anotacions que el delegat ha fet fins al moment.

## 4. DISSENY

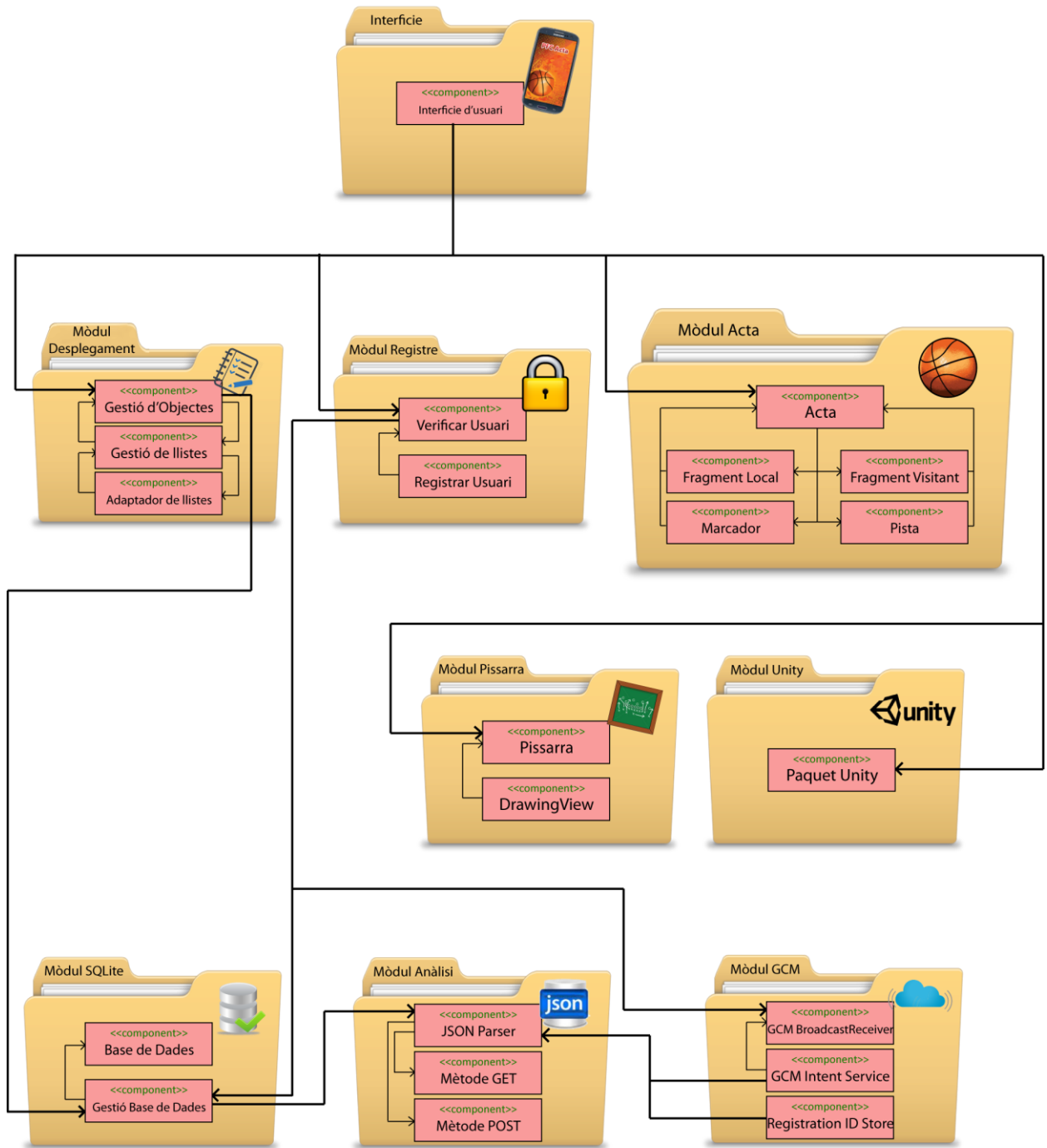
### 4.1 Diagrama detallat

Per a dissenyar un diagrama detallat i establir els diferents elements que hi apareixeran, és necessari realitzar una divisió a alt nivell de l'aplicació. Aquesta pot estar dividida en diferents mòduls encarregats de les principals funcionalitats que s'hauran de dur a terme.



**Figura 15:** Diagrama global de mòduls de l'aplicació.

La següent figura correspon al diagrama de components de l'aplicació. Aquí es representen els principals components dels quals contarà el sistema i les relacions entre ells per a desenvolupar la feina desitjada.



**Figura 16:** Diagrama detallat de l'ús dels mòduls de l'aplicació.

## 4.2 Interfície d'usuari

Per a dissenyar la interfície d'usuari ens hem basat en els requeriments de cada usuari i en els diagrames anteriors per a desenvolupar uns esbossos que ens serviran de guia alhora d'implementar la part gràfica de la nostra aplicació. Amb aquests esbossos tindrem una primera visió del que serà l'aplicació i ens ajudaran a discutir sobre quina és la millor estratègia alhora de mostrar tota la informació. Com que només serà una guia per al disseny final, no ens centrarem essencialment en la composició, ja que aquesta potser varia depenent de les restriccions funcionals de la pantalla en si.

Comencem doncs amb els diferents dissenys de cada pantalla a visualitzar:

### 4.2.1 Pantalla de logueig/registre

La pantalla de logueig és la pantalla inicial de l'aplicació, on deixem que l'usuari faci el *login* amb les seves credencials. Aquesta pantalla només ha de deixar escriure el nom d'usuari i contrasenya per a poder accedir a l'aplicació, anar a la pantalla de registre, o sortir de l'aplicació.

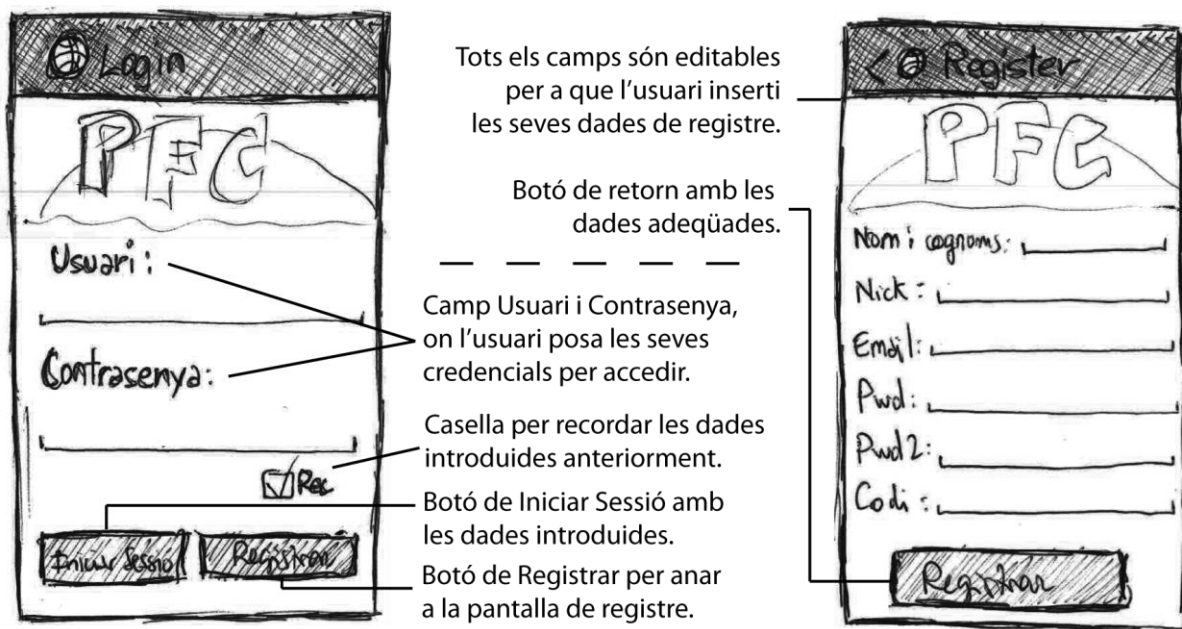
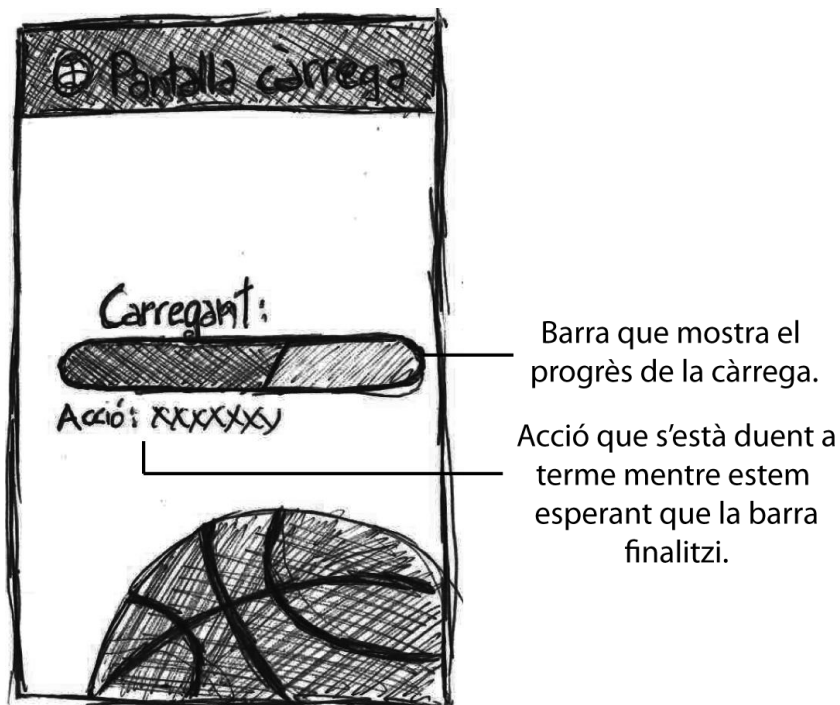


Figura 17: Esquema bàsic de la pantalla de logueix.

## 4.2.2 Pantalla de càrrega

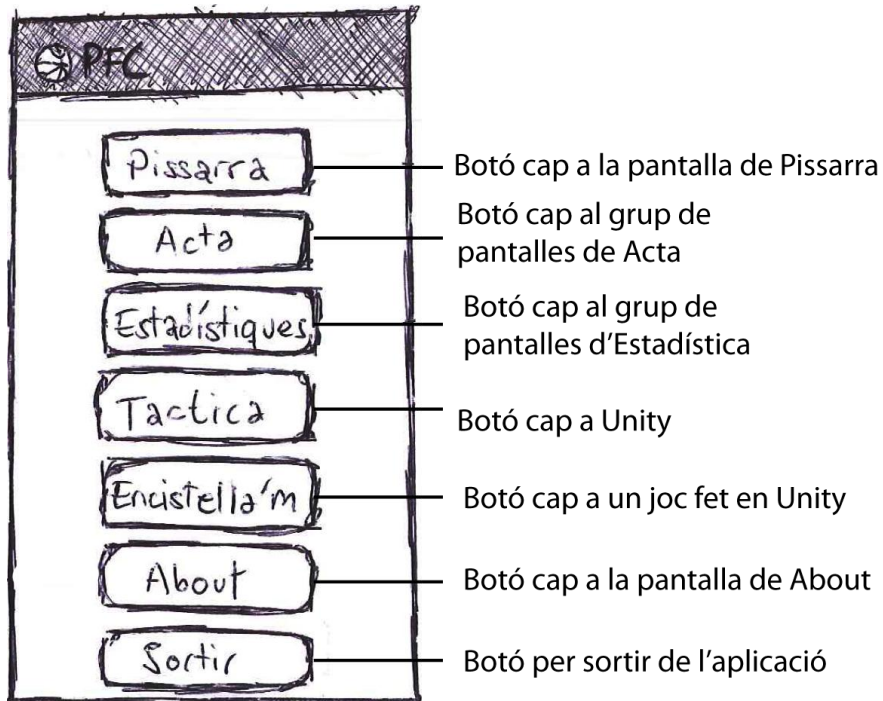
És la pantalla de transició entre el logueig i la pantalla principal de l'aplicació. Com normalment totes les funcions d'accés al servidor solen retardar una mica el funcionament normal de l'aplicació, he preferit crear una pantalla d'inici que emmascarés totes les accions a dur a terme darrere una barra de progrés.



**Figura 18:** Esquema bàsic de la pantalla de càrrega.

### 4.2.3 Pantalla de menú principal

La pantalla principal de l'aplicació és on cada usuari escollirà les diferents funcionalitats que necessita en cada moment, per tant, només contindrà botons per accedir a les diverses funcions.



**Figura 19:** Esquema bàsic de la pantalla de menú principal.

## 4.2.4 Pantalla de pissarra

És una pantalla final i no de transició com les altres. Aquí hi desenvolupem una pantalla on tota la superfície d'aquesta ha de poder ser pintada per l'usuari amb el dit o amb algun llapis digital, a més de contenir diferents botons que modifiquin la funció del dit (pintar, esborrar, fer la punta més fina...).



**Figura 20:** Esquema bàsic de la pantalla de pissarra.



### 4.2.5 Pantalla d'acta

Aquesta també és una pantalla final, tot i que per arribar a enregistrar una acta haurem de passar per varies pantalles prèvies per a indicar quin tipus de partit s'està jugant: com la categoria o l'equip local i visitant. Abans hi haurà una pantalla de consulta de les actes prèviament fetes o crear una acta nova.

Aquestes pantalles, en no ser funcionalment complexes o decisives, les posem dins d'aquest apartat anomenant-les *Pantalles Pre-Acta*.

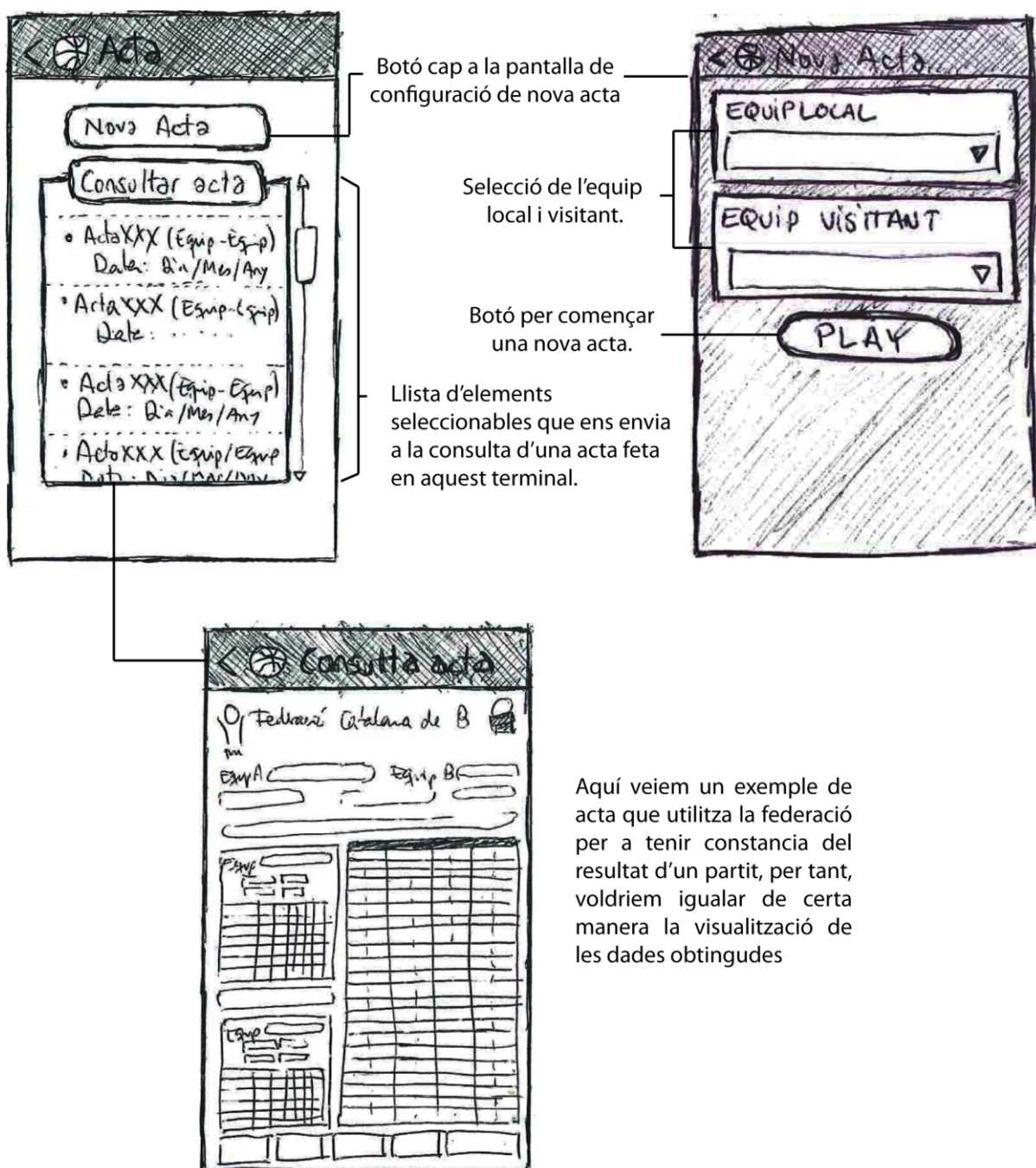
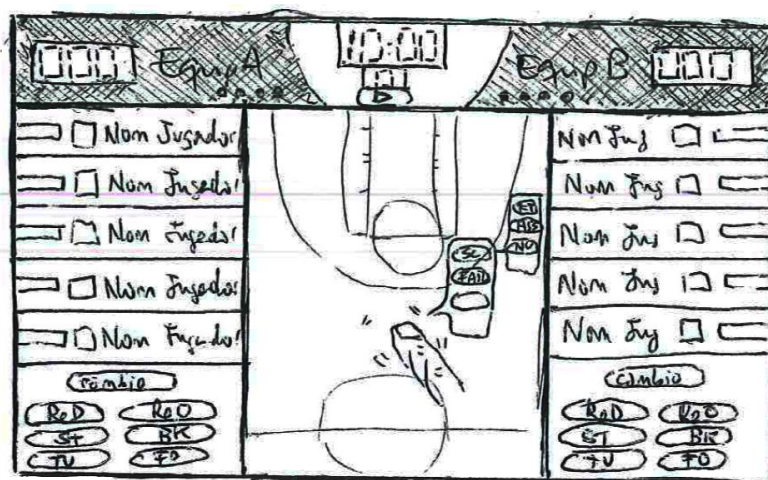


Figura 21: Esquema bàsic de la pantalla de pre-acta.

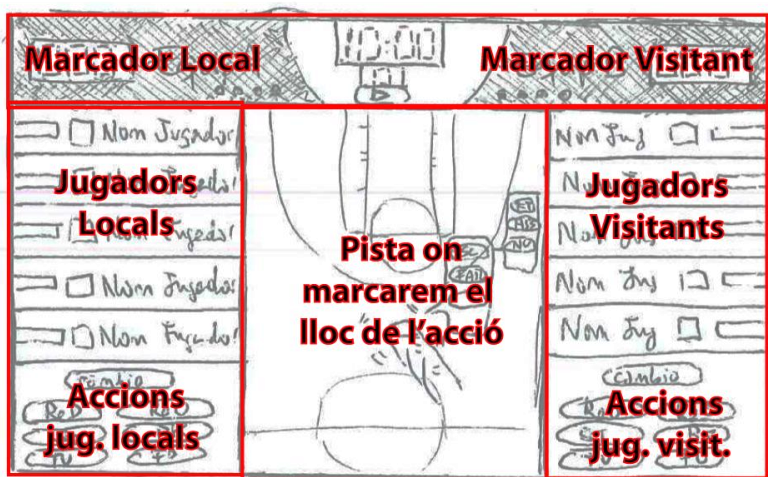
La complexitat de la pantalla d'acta fa dissenyar diverses maneres d'enregistrar cada esdeveniment. Tot i tenir clares les funcions que s'han d'implementar, la manera de presentar-les pot ser tan variada com complexa. He preferit fer un disseny senzill per poder arribar a qualsevol usuari i no endinsar-me en dissenys més complicats que només un usuari expert podria utilitzar en la seva totalitat.

Per tant, havent de fer un disseny amigable, la pantalla es presentarà en diverses seqüències de botons que ens aniran guiant a través del desenvolupament de l'acta. Per tal d'utilitzar-lo correctament, deixarem desconnectades les opcions no permeses.

Així doncs el disseny d'aquesta pantalla quedaria provisionalment d'aquesta manera:



Diferents parts de la pantalla están situades de manera que sigui intuïtiu la interacció.



Podem veure que la pantalla està dividida en 4 parts diferenciades:

- Marcador
- Part Local
- Part Visitant
- Pista

Figura 22: Esquema bàsic de la pantalla d'acta.

### 4.2.6 Pantalla d'estadística

La pantalla d'estadística hauria de ser una pantalla final, però el fet de poder consulta diferents estadístiques, complica la pantalla i per això he considerat que seria millor tenir una vista com a grup de pantalles. Aquí farem que les estadístiques de cada jugador o equip es puguin visualitzar individualment, per partit o globalment.

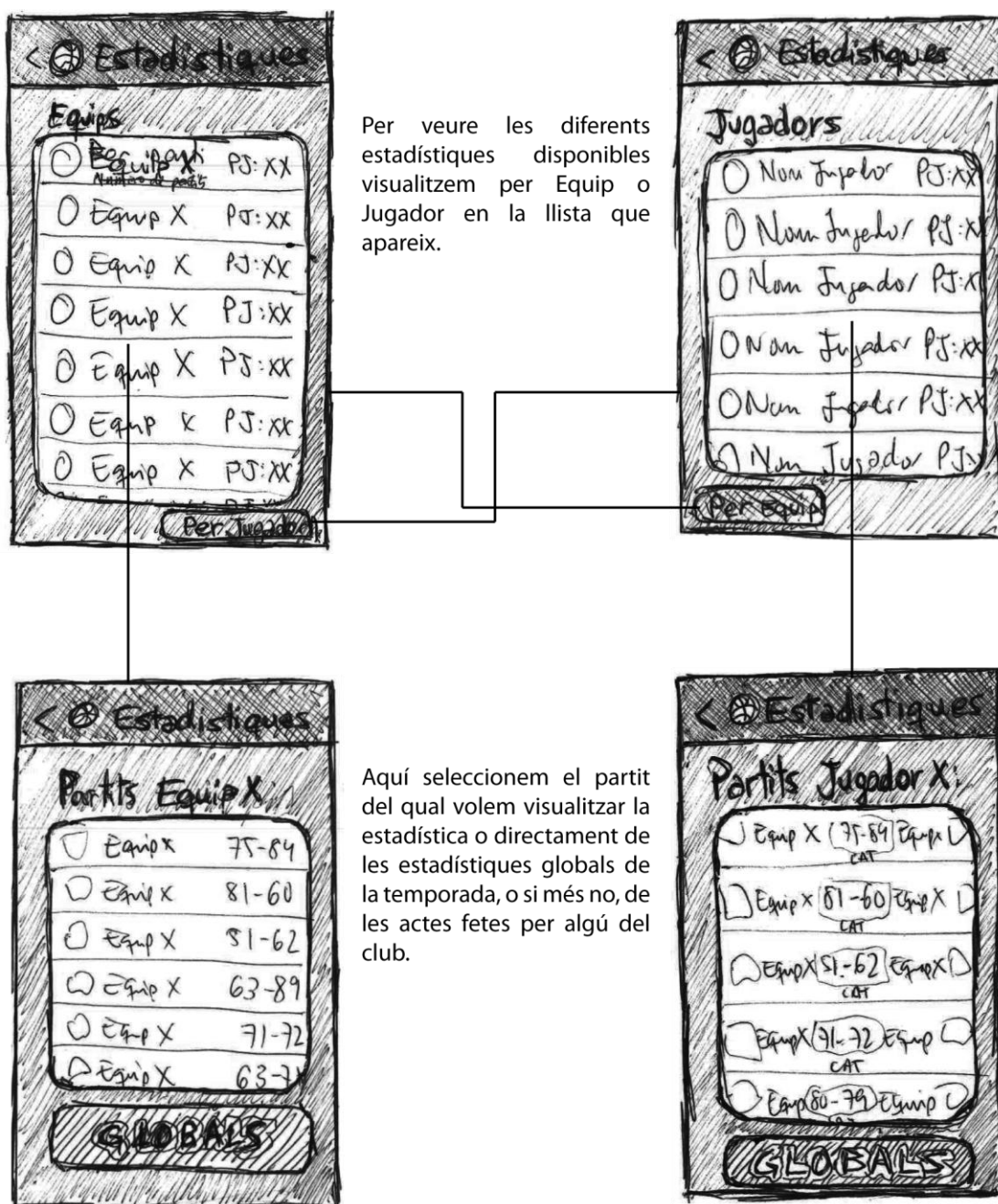
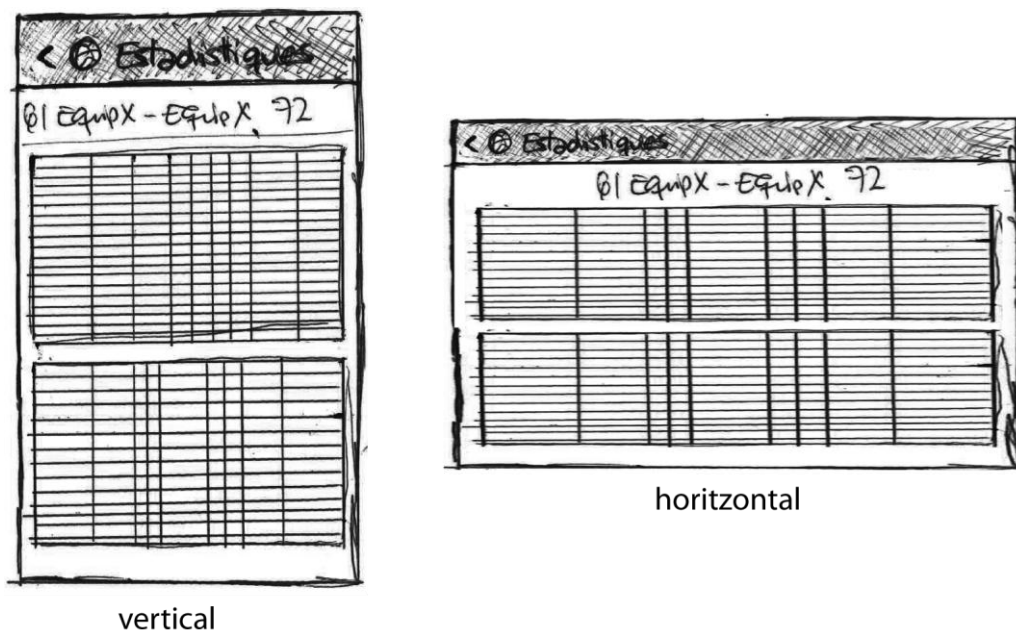


Figura 23: Esquema bàsic de la pantalla d'estadística.

Per visualitzar millor la part de les estadístiques permetem que la pantalla pugui estar també en horitzontal.



**Figura 24:** Esquema bàsic de la pantalla d'estadística final.

## 4.2.7 Pantalla de tàctica

La pantalla de tàctica engloba diferents pantalles amb varies funcionalitats, però els seus elements no varien gaire entre si. Com aquest grup de pantalles són les úniques que estan dissenyades amb una altra plataforma que no és Android, canviarem una mica el disseny en comparació amb les altres per adaptar-lo a l'editor Unity.

Per tant, la pantalla de tàctica la podem considerar com una pantalla principal dins del grup, on les altres faran les funcions de pantalla final. Així que la pantalla de tàctica ens quedaria:

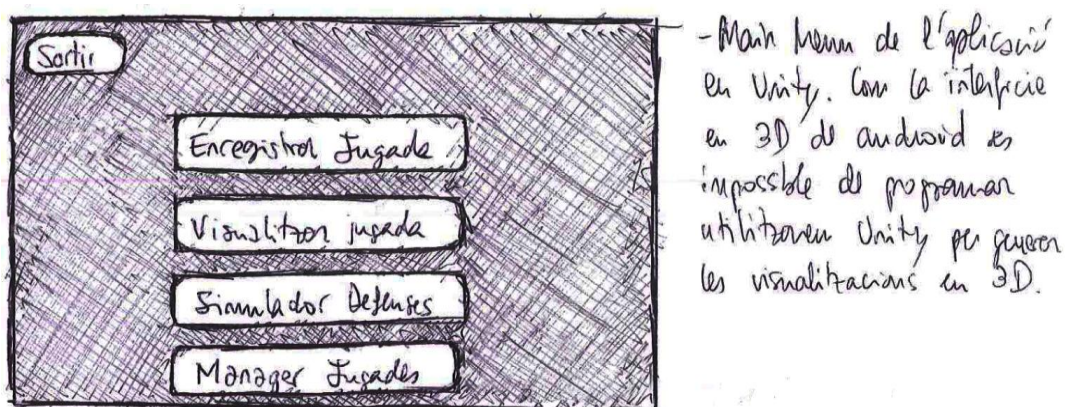
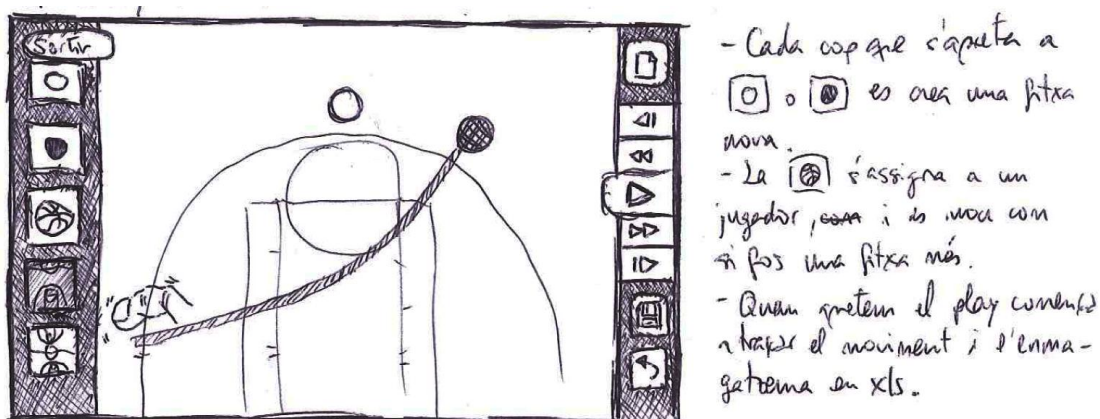


Figura 25: Esquema bàsic de la pantalla de tàctica.

### 4.2.7.1 Pantalla d'enregistrar jugada

La pantalla d'enregistrar jugada és possiblement la més complexa d'aquest grup, ja que en ella l'usuari ha de poder muntar una jugada pas per pas. Tal i com passava amb la pantalla d'acta, aquesta pantalla per la seva complexitat, també és difícil de crear a causa de la gran quantitat de possibilitats que hi poden haver alhora de crear un moviment que ha d'anar compassat amb d'altres. Per aquest motiu, s'ha tractat la pantalla com un seguit de passos o "steps", en què l'usuari, a cada pas, pot fer tots els moviments que s'han de sincronitzar amb d'altres, sense que aquests moviments es des-sincronitzin.



**Figura 26:** Esquema bàsic de la pantalla d'enregistrar jugada.

### 4.2.7.2 Pantalla de reproduir jugada

La pantalla de reproduir jugada és únicament la reproducció del moviment de la pantalla anterior, per tant aquesta únicament constarà d'un controlador de passos i d'un botó d'inici.

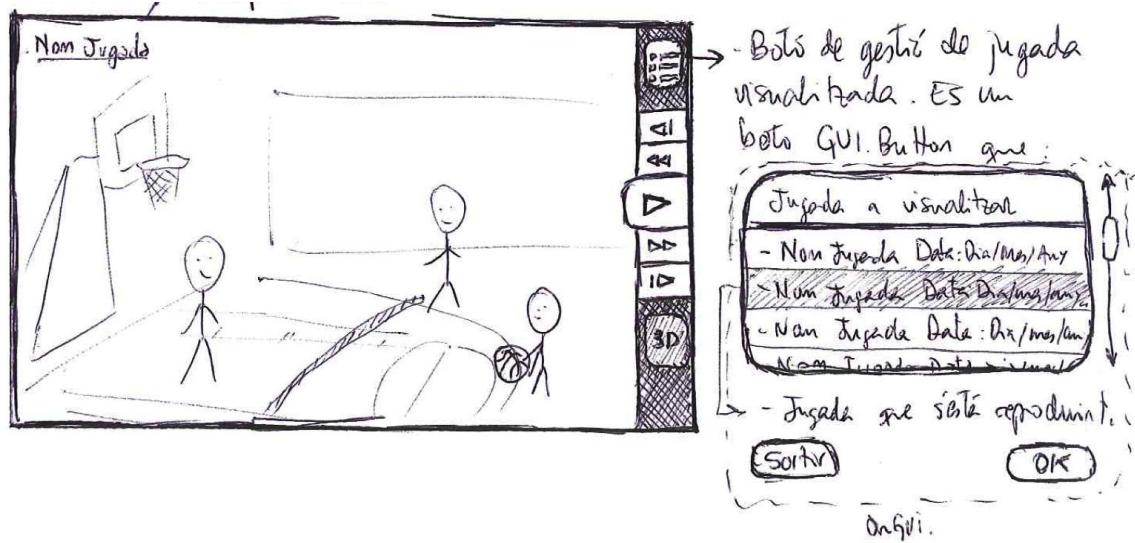


Figura 27: Esquema bàsic de la pantalla de reproduir jugada.

### 4.2.7.3 Pantalla de simulació de defenses

La pantalla de simulació de defenses és un grup de pantalles formada per dues pantalles. En una s'hi posa l'atac i la defensa a simular, i en l'altra es visualitza l'evolució de la simulació en quan als dos paràmetres establerts.

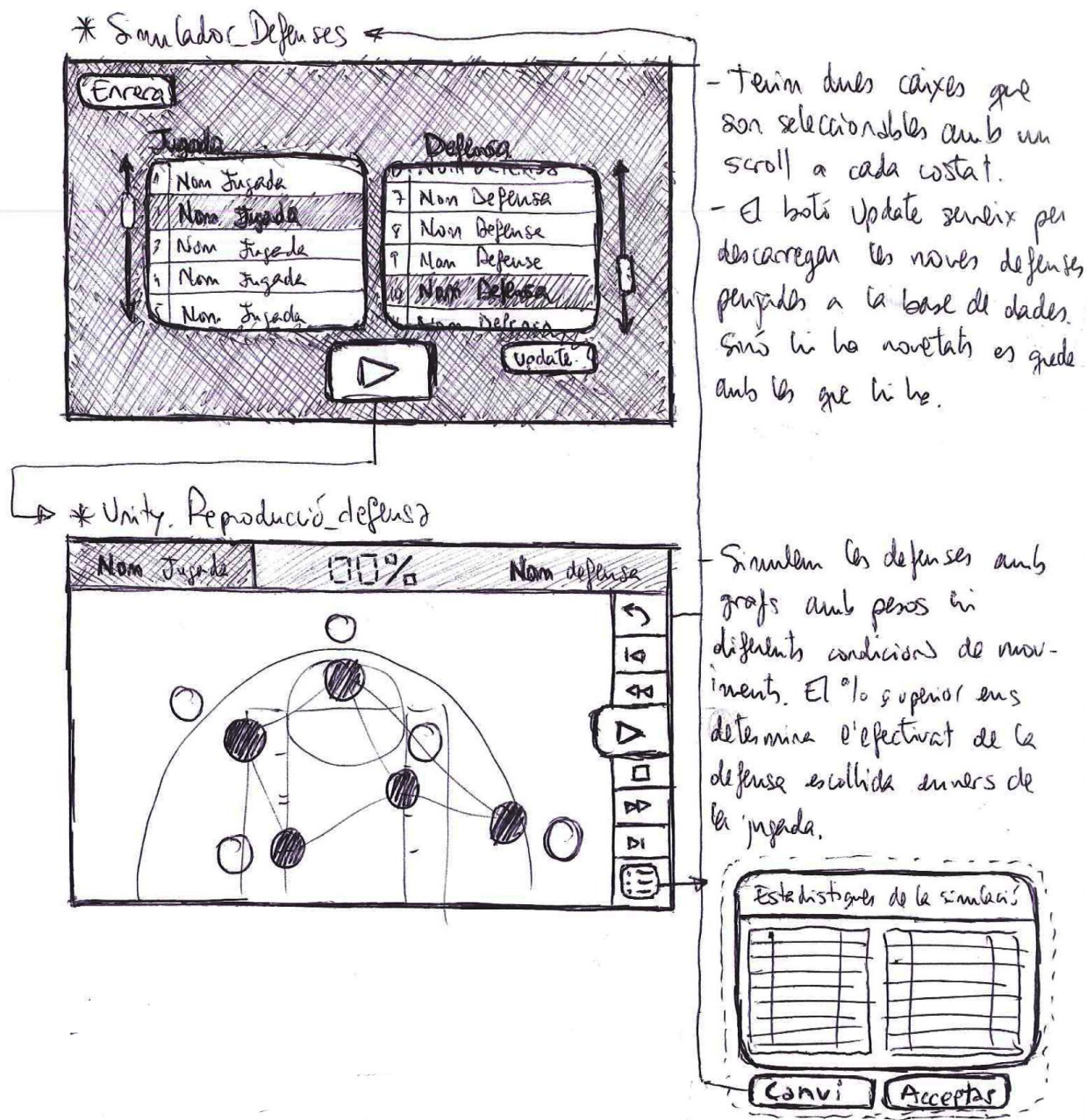


Figura 28: Esquema bàsic de la pantalla de simulació de defenses.



### 4.2.7.4 Pantalla de manager de jugades

Aquesta pantalla és com un gestor de jugades, és a dir, aquí podem esborrar, compartir o descarregar les jugades que hem creat o hagin creat d'altres usuaris.

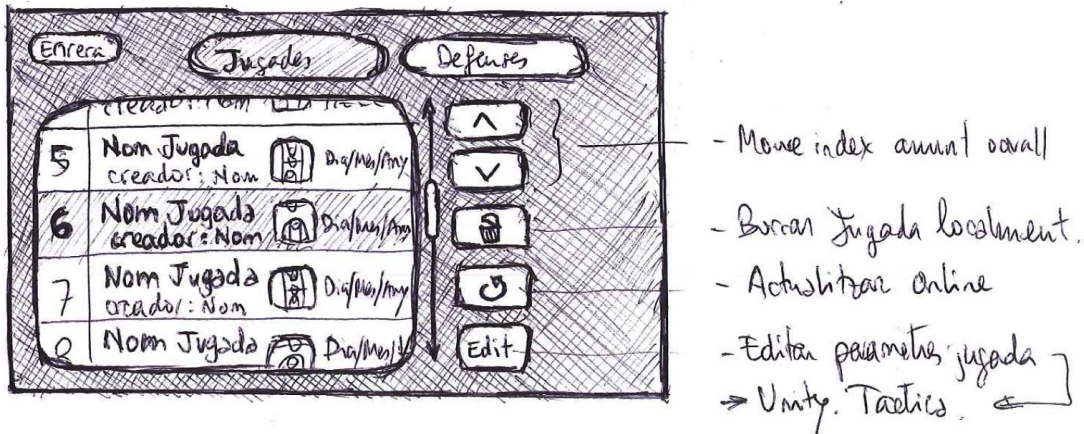


Figura 29: Esquema bàsic de la pantalla de manager de jugades.

## **5. IMPLEMENTACIÓ**

### **5.1 Base de Dades SQL**

Un dels objectius fonamentals d'un sistema d'informació és comptar no només amb els recursos d'informació, sinó també amb els mecanismes necessaris per a poder trobar i recuperar aquests recursos. D'aquesta manera, les bases de dades s'han convertit en un element indispensable no només per al funcionament dels grans motors de cerca i la recuperació d'informació al llarg i ample de la web, sinó també per a la creació de les seues web, intranets i altres sistemes d'informació en els quals es precisa gestionar grans i petits volums d'informació. La creació d'una base de dades a la que puguin acudir els usuaris per fer consultes i accedir a la informació que els interessa és, doncs, una eina imprescindible de qualsevol sistema informàtic/aplicació.

Una base de dades és una col·lecció de dades organitzades i estructurades segons un determinat model d'informació que reflexa les dades en sí mateixes, així com les relacions que existeixen entre elles. Una base de dades es dissenya amb un propòsit específic i ha d'estar organitzada amb lògica coherent. Les dades poden ser compartides per diferents usuaris i aplicacions, però han de conservar la seva integritat i seguretat al marge de les interaccions d'ambdós. La definició i descripció de les dades han de ser úniques per a minimitzar la redundància i maximitzar la independència en la seva utilització.

En una base de dades, les entitats i els atributs del món rea, es converteixen en registres i camps. Aquestes entitats poden ser tant objectes materials com llibres o fotografies, però també persones i inclús conceptes i idees abstractes. Les entitats posseeixen atributs i mantenen relacions entre elles.

#### **5.1.1 Creació de la base de dades**

Dins del marc de l'aplicació que desenvolupem he creat la base de dades que s'adequa a les necessitats i funcionalitats que regiran una estructura coherent i no redundant. Veiem aquí el diagrama d'entitat-relació que s'ha creat per a l'aplicació:

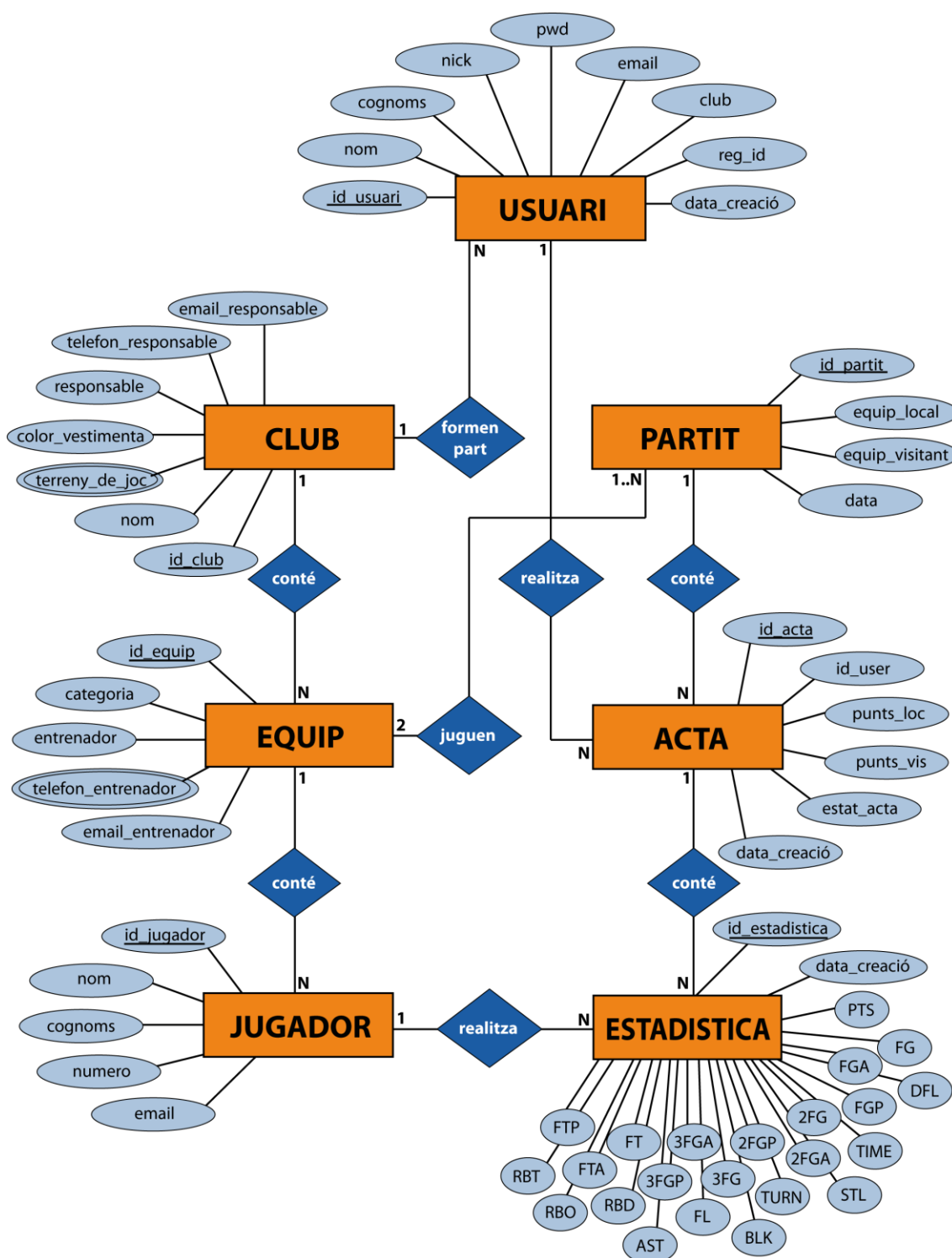


Figura 30: Diagrama E-R de la base de dades utilitzada.

Amb aquestes entitats i relacions podem crear una base de dades senzilla però complerta. El fet de tenir poques entitats, facilita molt la comprensió i gestió d'aquestes a l'hora de fer consultes o verificar la coherència de les dades que s'introdueixen. Però el fet d'haver-hi més relacions que entitats, ens complica una mica l'obtenció de les dades en les consultes, ja que seran consultes més exhaustives, perquè s'haurà de consultar resultats parcials (entre taules) segons la consulta que es

requereix-hi. Així que amb aquest diagrama podem obtenir ja la base de dades funcional amb SQL Developer.

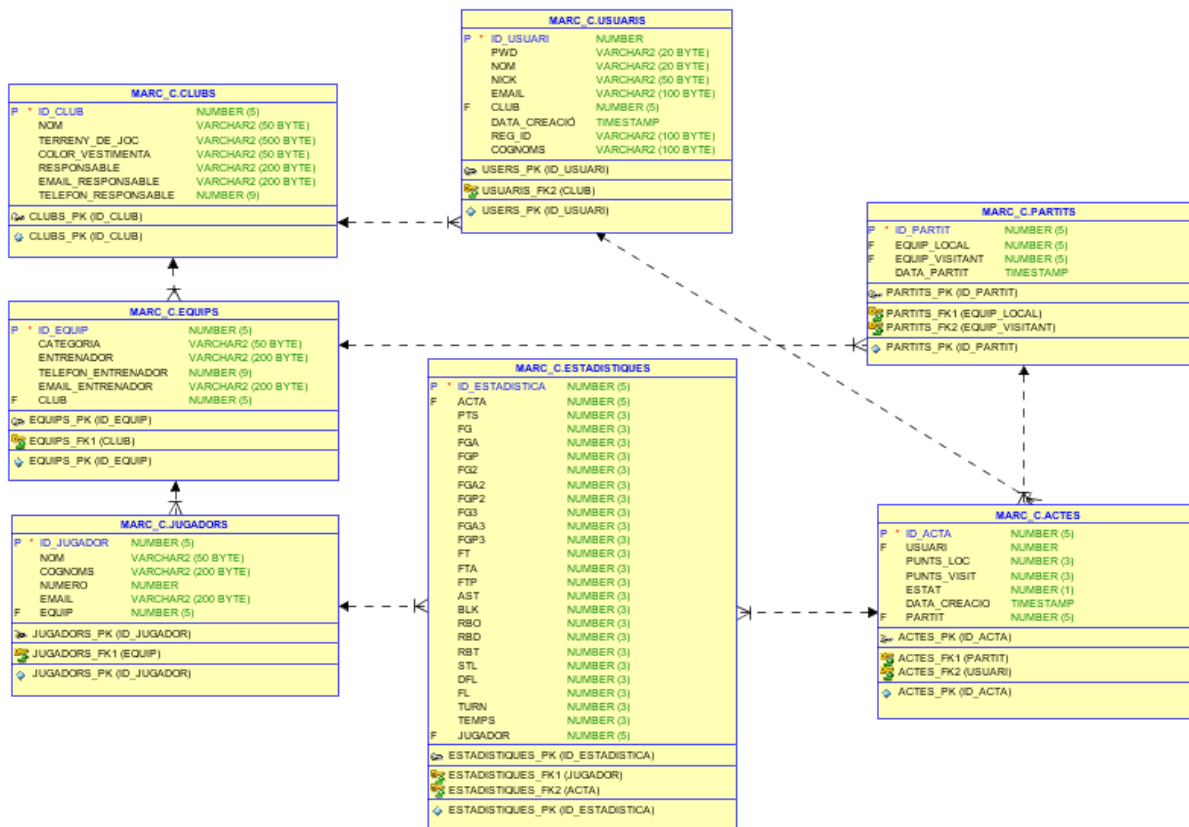


Figura 31: Diagrama E-R de la base de dades generada per Oracle SQL Developer.

### 5.1.2 Creació de php's per a la consulta

Primerament introduïrem què és el PHP, ja que encara no hem vist que ens aporta a l'hora de desenvolupar el projecte.

PHP (Hypertext Pre-Processor) és un llenguatge de programació utilitzat generalment en la creació de continguts per a pàgines web. És un llenguatge interpretat que s'utilitza especialment per a crear contingut web dinàmic i aplicacions per a servidors, tot i que també és possible crear aplicacions gràfiques utilitzant GTK+.

Si bé existeixen alternatives per a tots els gustos, molts consideren PHP com l'eina indispensable a l'hora de desenvolupar un lloc web. Generalment els scripts en PHP s'incrusten en codis com HTML, ampliant enormement les possibilitats del dissenyador de pàgines web.

La interpretació i execució dels scripts PHP es fan en el servidor, el client (un navegador que demana una pàgina web) només rep el resultat de l'execució i mai es veu el codi PHP que hi ha darrera.

Permet la connexió a tot tipus de servidors de base de dades, que és la característica que ens interessa per desenvolupar l'aplicació, com MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird i SQLite.

Així doncs comencem a definir quins scripts necessitarem per a crear les accions a la base de dades per a obtenir el resultat correcte en cada cas:

- Configuració: script de configuració on hi haurà totes les dades necessàries de configuració de la connexió del dispositiu mòbil amb la base de dades. Aquest arxiu no conté cap funció, únicament serà de consulta per als altres scripts.
- Connexió: script de connexió amb la base de dades. Aquí tindrem les funcions necessàries per a que altres scripts puguin fer la correcta connexió amb la base de dades. Amb aquest script ens estalviem la reescriptura en cada script que necessiti connectar-se a la base de dades.
- Registre: script per a la inserció d'un nou usuari a la base de dades, on consultarà si les dades enviades són correctes per la inserció o ja hi ha un usuari registrat amb les credencials enviades.
- GCM: script necessari per a la connexió amb el Google Cloud Messaging, que permet que un usuari que ha fet una acció determinada per a altres usuaris sigui registrada i conseqüentment enviada la notificació als dispositius requerits. Amb això ens estalviem que els usuaris estiguin constantment fent consultes a la base de dades per a verificar si hi ha informació nova per a descarregar, cosa que faria l'aplicació molt ineficient.
- Funcions: script que contindrà tots els tipus de consultes a la base de dades per a obtenir el resultat sol·licitat per l'aplicació android. Segons els paràmetres d'entrada, realitzarem l'acció necessària per a obtenir el valor que necessitem.

## 5.2 Android

En aquest apartat descriurem com es comença a desenvolupar l'aplicació a partir de zero. Tal i com l'ordre d'evolució que hem seguit en l'apartat de "disseny", seguirem el mateix en la implementació, ja que ens servirà per anar pas a pas.

### 5.2.1 Pantalla de logueix

Un cop executem l'aplicació, la primera pantalla que ens trobarem serà la de loguejar-se. L'usuari ha de poder introduir les seves dades en les caixes de text editable, juntament amb un *Checkbox* que ens permeti guardar la informació introduïda per a futures execucions a través de les preferències compartides. A més, també hauria d'haver-hi un botó que enviés l'usuari a la pantalla de registre per a poder-ne crear un de nou, en cas de no tenir-ne, i així, poder gaudir de l'aplicació.

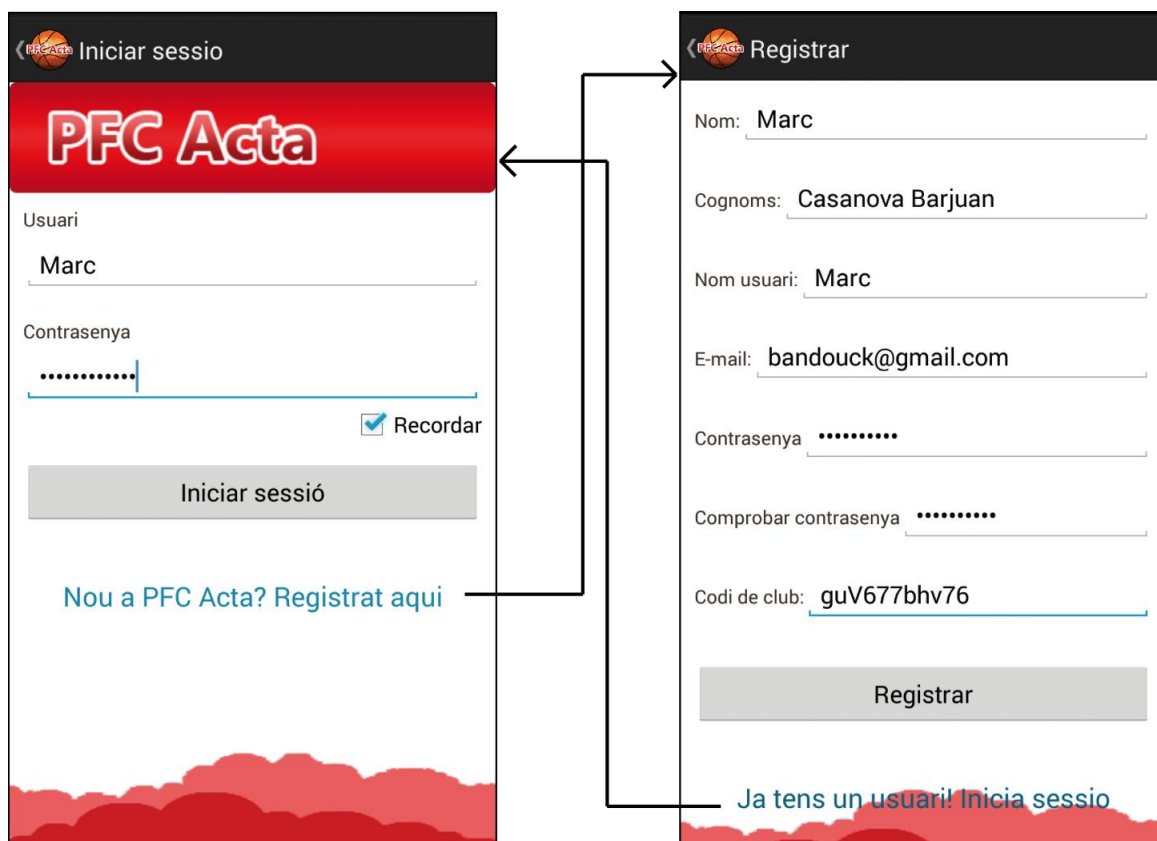
La part funcional d'aquesta pantalla és la verificació en background de l'usuari al servidor. Un cop introduïdes les credencials, l'aplicació enviarà a través d'una tasca asíncrona una petició de confirmació de credencials al servidor, on el php de verificació d'usuari que hem vist anteriorment farà la consulta SQL i retornarà si l'usuari és vàlid. Un cop validat l'usuari i guardat en les preferències de l'aplicació, en un segon logueix si l'usuari és el mateix que el guardat anteriorment, no caldrà que faci la connexió al servidor i es loguejarà automàticament.

De diferent manera, quan no tenim connexió a la xarxa aquestes connexions no es poden establir i no tindrem capacitat per iniciar l'aplicació, per tant, és necessari iniciar-la per primer cop amb connexió en xarxa, atès que després ja es podrà accedir amb el mateix usuari. Això és deu a que l'aplicació guarda en les seves preferències els usuaris ja loguejats amb anterioritat.

Una altra funcionalitat d'aquesta pantalla de logueix és poder anar a altres pantalles com: la pantalla de registre, la pantalla principal, o , si s'escau, sortir de l'aplicació. Si les Activitats són bàsicament pantalles, els *intents* són la manera d'invocar aquestes pantalles. "Un intent és la descripció abstracta d'una operació que s'ha de dur a terme". O dit d'una altra manera, un *intent* és una classe que permet especificar una Activitat a executar. Per tant, necessitarem diferents intents associats als botons de la pantalla per a poder moure'ns entre pantalles o Activitats.

La pantalla de registre és molt similar a la de logueix, amb l'única diferència funcional que permet introduir informació nova a la base de dades. Un cop l'usuari ha introduït totes les seves dades per a registrar-se i ha premut el botó de registrar, s'executarà la tasca de verificar l'usuari, igual que en la de logueix, però únicament per confirmar que l'usuari que es vol introduir no existeix prèviament. Un cop obtinguda la resposta, es crida a una tasca asíncrona de "registrar usuari", que introdueix les dades a la base de dades per tal de poder accedir a l'aplicació.

La interfície gràfica d'aquestes dues pantalles és la següent:



**Figura 32:** Esquema original de la pantalla de logueix.

## 5.2.2 Pantalla de càrrega

Un cop fet el logueix trobarem una pantalla de transició que ens permetrà fer les accions necessàries en la configuració de l'aplicació. Aquesta pantalla de càrrega només consta d'una tasca a executar, que és l'actualització, si cal, de la base de dades interna de l'aplicació si hi ha hagut algun canvi d'usuari. Això ens permetrà gestionar els equips del nostre club, sense interactuar directament amb les dades d'altres clubs.

Funcionalment només s'executa una tasca de descarrega d'informació de la base de dades, així que la velocitat en l'execució depèn únicament de la connexió que tinguem en aquest moment.



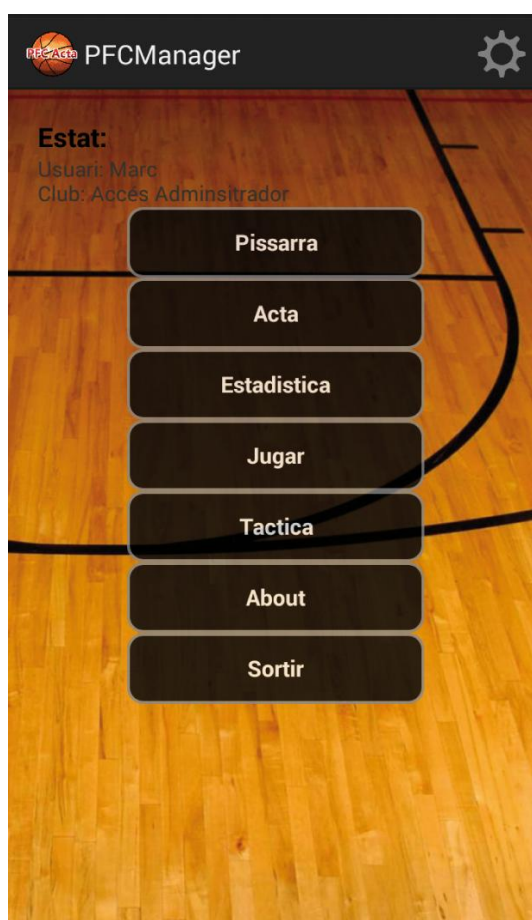
**Figura 33:** Esquema original de la pantalla de càrrega.



### 5.2.3 Pantalla principal

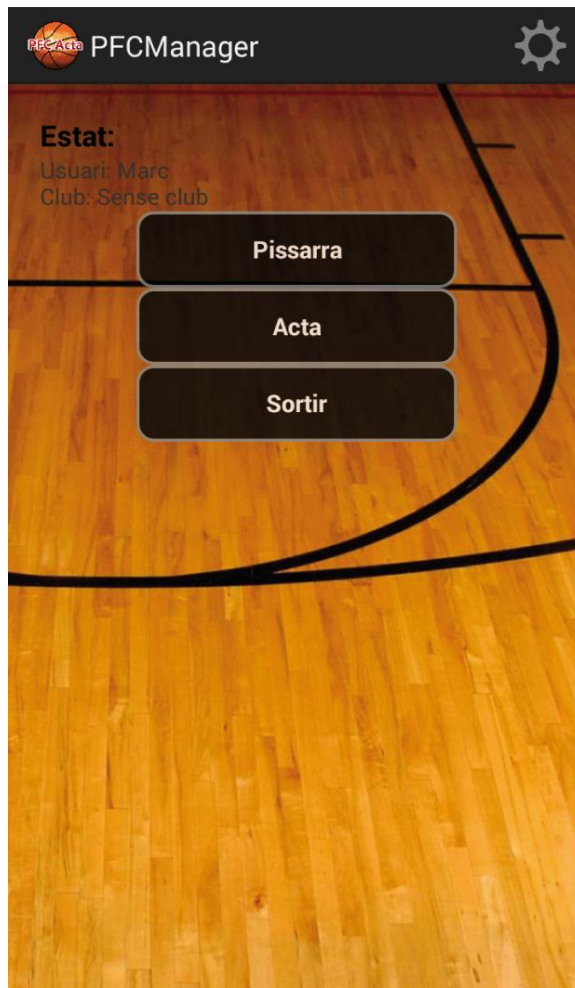
Un cop s'ha carregat la base de dades, veiem la pantalla principal. En aquesta pantalla hi trobem diferents botons que ens duran a les pantalles finals que necessitem. També apareix un text que ens diu quin tipus d'usuari som, ja que depenent d'aquest tipus ens permetrà accedir a diferents pantalles o bloquejar-nos l'accés.

Funcionalment aquesta pantalla és només de transició cap a pantalles finals, per tant, només estarà compresa per *intents* cap a d'altres pantalles.



**Figura 34:** Esquema original de la pantalla de menú principal sense restriccions.

Ara bé, si el nostre nivell d'accés és restringit, és a dir, si no formem part d'un club (o no tenim el codi d'accés) aquesta pantalla restringirà l'accés als diferents apartats fets, únicament, per a la gestió interna del club. Per tant, haurem de fer que la visibilitat d'alguns botons desaparegui.



**Figura 35:** Esquema original de la pantalla de menú principal amb restriccions.

## 5.2.4 Pantalla de pissarra

La pissarra és un element bàsic per a qualsevol entrenador, ja que gràficament, és ràpid i senzill donar indicacions precises als jugadors a l'hora de fer una jugada.

Aquesta pantalla està formada per dues Activitats diferents, una que controla el marc de l'execució i els botons que es polsen i l'altre que detecta i dibuixa sobre la pantalla/vista. Amb això aconseguim tenir un menú on podem decidir el tipus de punter que volem utilitzar, variant entre forma mida i funció, a més d'altres funcions de configuració de la pissarra.

Els diferents botons que hi ha són:



**Figura 36:** Barra d'accions/eines de la pissarra

- Botó de pissarra nova, sempre en una aplicació de dibuix hi ha d'haver una forma rapida d'esborrar i obtenir un nou marc de dibuix.



**Figura 37:** Botó de nova pissarra.

Amb aquest botó obrirem un diàleg per a saber si es vol guardar abans d'esborrar, per si s'hagués polsat el botó sense voler.



**Figura 38:** Diàleg quan polsem el botó de pissarra nova

- Botó de mida de pinzell, per a decidir quin és el gruix de la línia que volem dibuixar.



**Figura 39:** Botó de canvi de mida de pinzell.

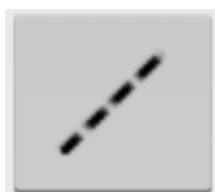
Amb una petita caixa de diàleg escollim entre tres mides diferents de



pinzell.

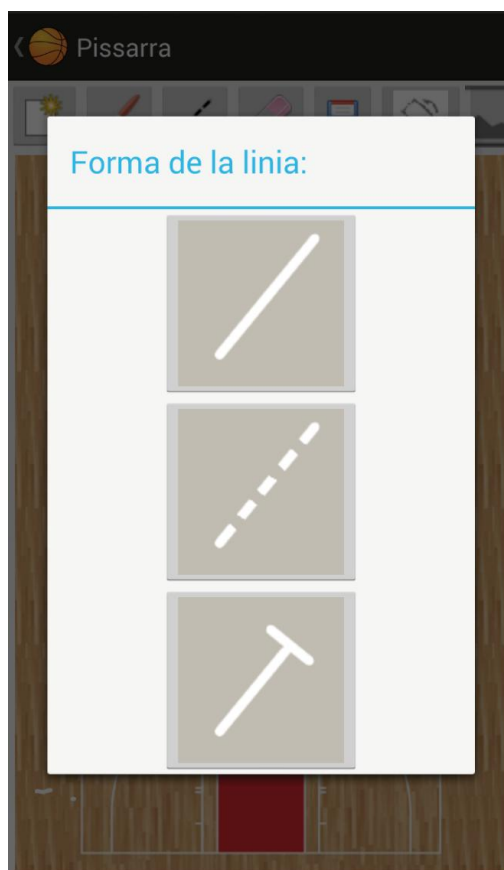
**Figura 40:** Diàleg quan polsem el botó de mida de pinzell

- Botó de forma de pinzell, per a definir quin serà el patró que seguirà la línia a dibuixar. Amb això podem canviar el significat de la línia en quan a la interpretació de la pissarra.



**Figura 41:** Botó de canvi de forma de traç.

Amb una petita caixa de diàleg escollirem les diferents formes que hi ha de fer una línia.



**Figura 42:** Diàleg quan polsem el botó de forma de traç.

- Botó de mida d'esborrador, idèntica a la de la mida del pinzell, però ara per definir la mida de la zona del esborrador.



**Figura 43:** Botó de canvi de pinzell per goma d'esborrar.

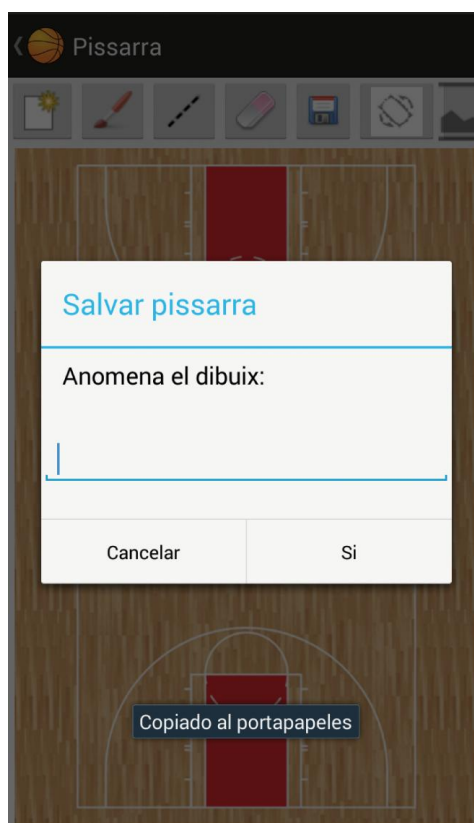
Amb la mateixa caixa de diàleg que la de mida de pinzell escollim entre tres mides diferents d'esborrador.

- Botó de guardar, per tal de poder guardar i visualitzar posteriorment qualsevol dibuix que s'hagi fet.



**Figura 44:** Botó de canvi de guardar.

Amb aquest botó obrirem un diàleg per a poder anomenar la imatge que hem fet a la pissarra.



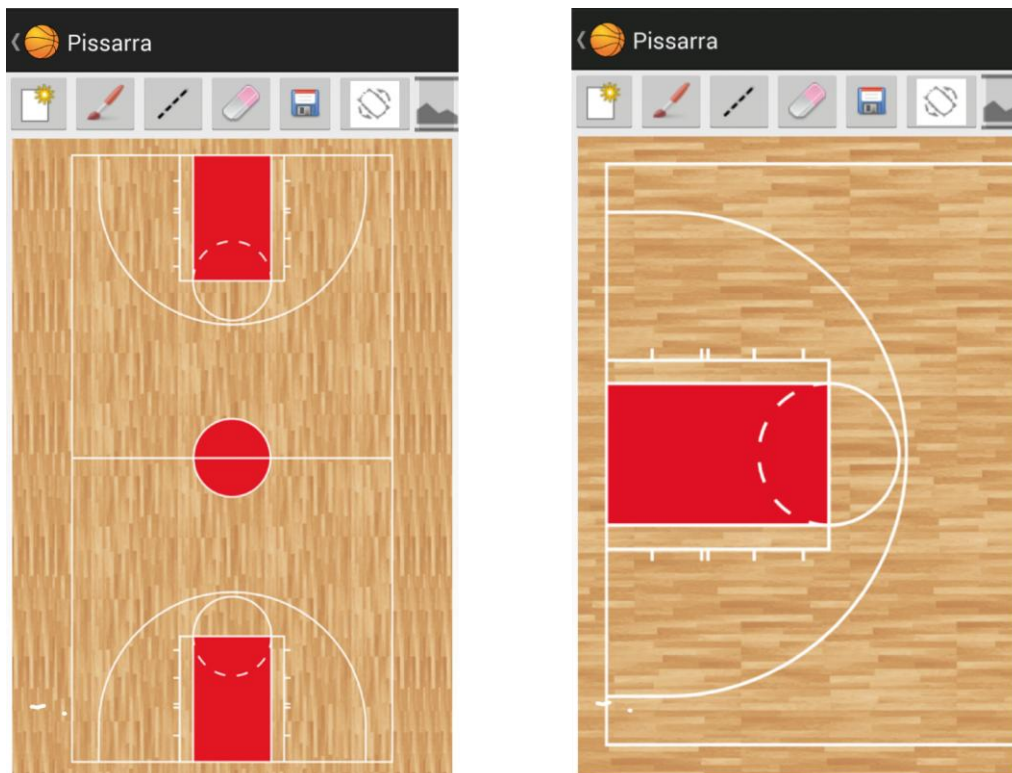
**Figura 45:** Diàleg quan polsem el botó de guardar.

- Botó de mitja pista, amb aquest botó podrem rotar el fons de la pissarra per adaptar-la a la jugada que ens interessa mostrar.



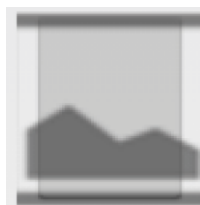
**Figura 46:** Botó de gir de pissarra.

Prement el botó aconseguim que el fons canviï de mitja pista a pista completa o al revés.



**Figura 47:** Representació de gir de pissarra.

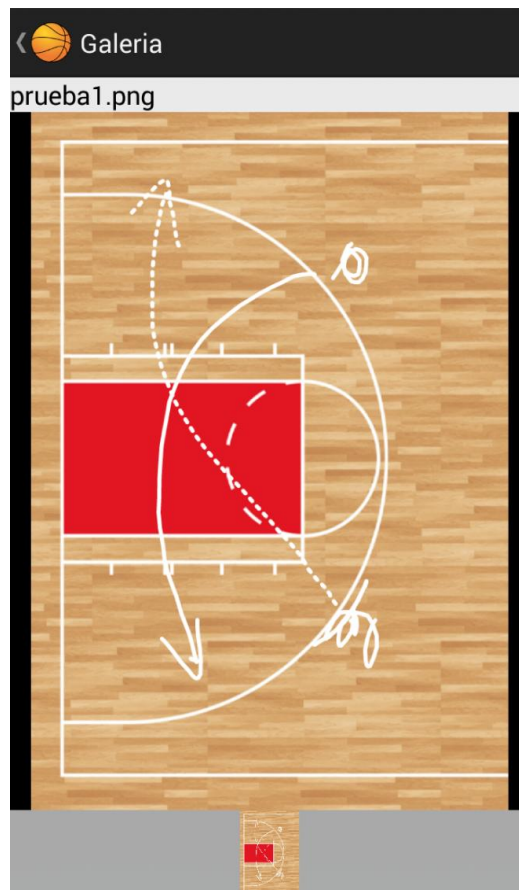
- Botó de galeria, aquest botó ens durà a una altra pantalla que en la versió de disseny no hi era. Això es deu a que posteriorment vaig preferir guardar les captures de pissarra dins de l'aplicació i no directament a la Galeria del dispositiu, ja que així és més còmode veure-les en l'aplicació directament i no anar a buscar-les sempre a la Galeria.



**Figura 48:** Botó de galeria.

Anirem directament a la pantalla de Galeria, on visualitzem les captures de pantalla guardades des d'una petita llista previsualitzada al peu de la pantalla i el nom que se li ha posat a sobre de la imatge.





**Figura 49:** Visualització de la galeria.

Concloem aquí l'apartat de pissarra i ens dirigim cap a Acta.

## 5.2.5 Pantalla d'acta

Aquesta pantalla és més complexa que l'anterior, ja que forma part d'un grup o seqüència de pantalles. Això es deu a que per a començar a crear una acta, haurem de establir certs paràmetres com és la categoria i els equips que s'enfronten.

El fet de ser un usuari sense club, provoca que hi hagi certs canvis a l'hora de desenvolupar l'acta, ja que no es podrà gaudir d'aquests paràmetres alhora de la configuració. Aquesta restricció és habitual en aquest tipus d'aplicacions, ja que fan que l'experiència de l'usuari no sigui del tot completa sense tenir els permisos adequats.

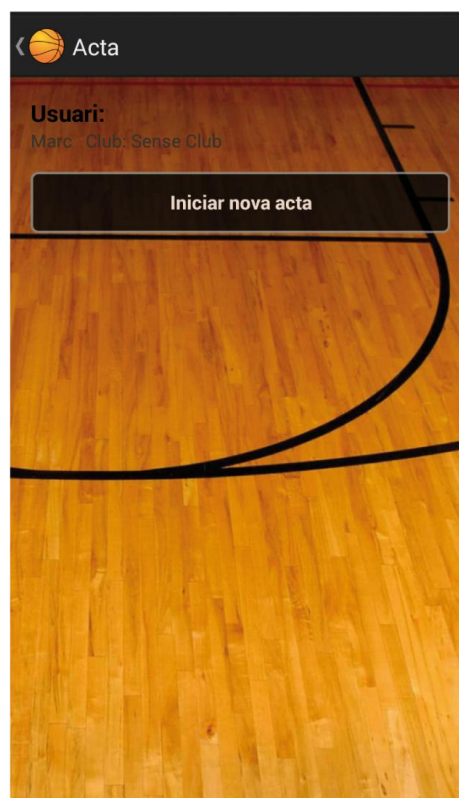
Així doncs, quan comencem a fer una acta haurem de fer aquests passos a través de diferents pantalles. En el cas de ser un usuari registrat i amb club, el primer que visualitzarem després de prémer el botó d'acta de la pantalla principal és una pantalla que ens deixa veure les actes fetes des de l'aplicació prèviament, juntament amb un botó per a la generació d'una nova acta que ens enviarà a la següent pantalla.

Funcionalment aquesta Pantalla consta de dues Activitats, una per al control dels elements de la pantalla i una altra que és un adaptador de dades, que permet visualitzar les dades en una llista personalitzada. Aquest fet provoca que hi hagi dues pantalles sobreposades, la principal i la llista, però que visualment estan sobreposades.

En canvi, si l'usuari no té club assignat, no es visualitzaran les actes fetes prèviament, i només es tindrà l'opció de crear una acta nova, com podem veure en la imatge següent:



Usuari amb club



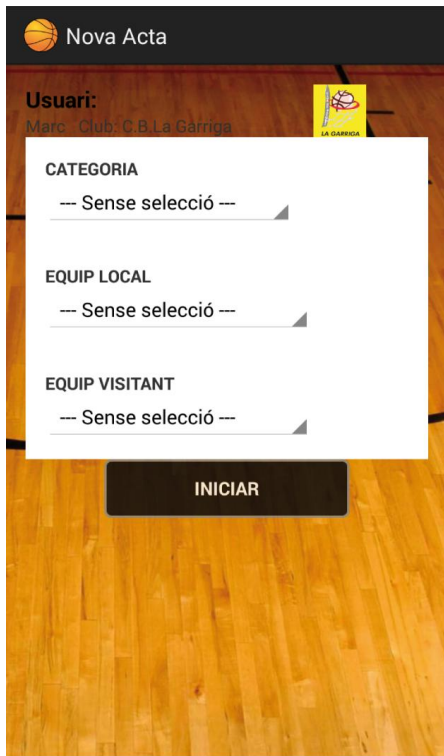
Usuari sense club

**Figura 50:** Esquema original de la pantalla de pre-acta.

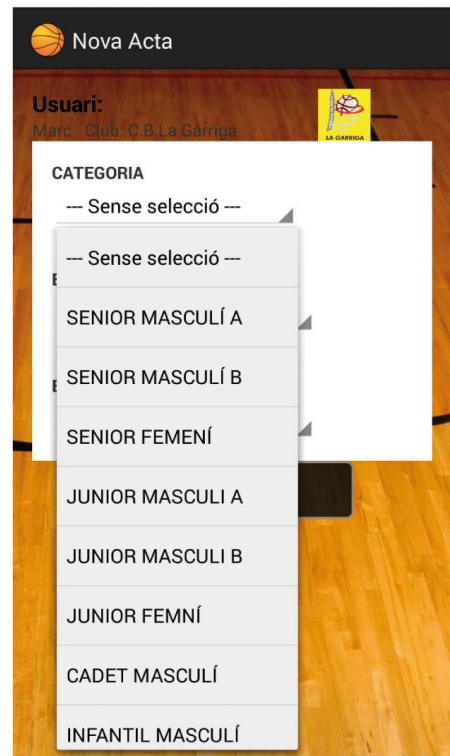
La següent pantalla de Nova Acta és més simple, únicament tindrem uns menús desplegable en què seleccionarem la categoria on juguen els equips i evidentment, els equips que hi participen. Aquí només es deixarà escollir l'equip del club al qual es pertany, tant sigui local com visitant i l'altre equip que participi en la mateixa categoria.

Funcionalment només s'ha de fer una cerca senzilla de la base de dades interna per saber quins són els equips formen part de la categoria seleccionada i introduir-los en els desplegable, per a que a cada categoria només es puguin seleccionar els equips que hi participen.

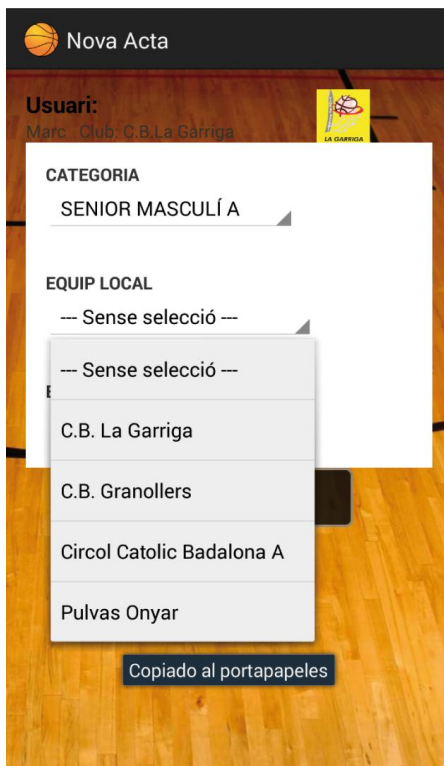
Aquesta pantalla s'elimina si l'usuari no té club, ja que no es permetrà introduir equips en l'acta, sinó que es farà una acta predefinida amb els números dels jugadors.



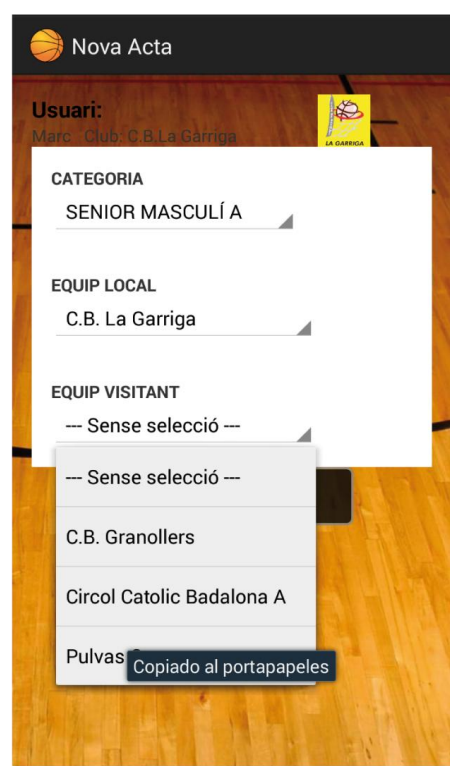
Pantalla al inici



Seleccionant categoria



Seleccionant equip local



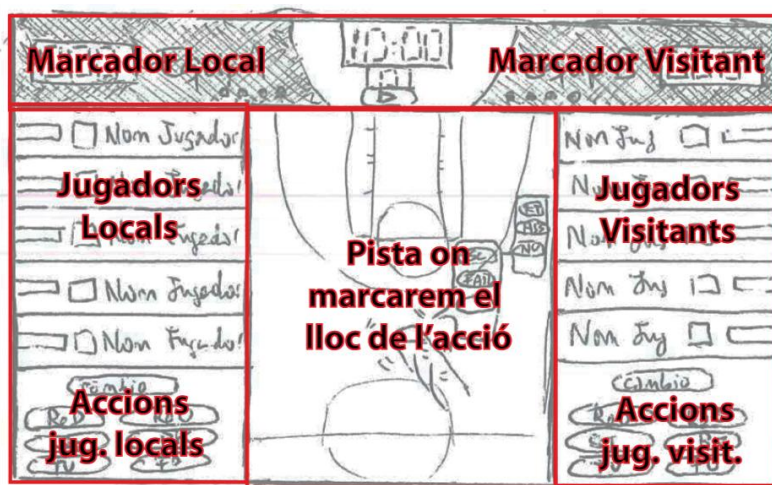
Seleccionant equip visitant

**Figura 51:** Esquema original de la pantalla de pre-acta 2.

Un cop arribem a la pantalla final de la seqüència, veiem que és més complexa que les anteriors, ja que s'ha de gestionar molta informació en un espai reduït predeterminat, i la dividirem en 5 parts, que són les 5 àrees delimitades que té l'acta:

- Controlador
- Equip local i equip visitant (funcionalment son idèntiques)
- Marcador
- Pista

Com em vist prèviament a l'apartat de disseny, aquestes àrees han estat posicionades de manera que l'usuari que vagi a fer l'acta, sàpiga en tot moment on ha d'anotar l'acció que s'ha produït en el terreny de joc.



Disseny previ



Disseny final

Figura 52: Esquema original de la pantalla de d'acta.

## Equip local i equip visitant:

Comencem doncs pels equips que simulen els jugadors que hi ha en pista, les accions simples (que estan bloquejades) i el botó de banqueta. Si polsem el botó d'algun jugador activem les accions simples i la pista (per començar a fer seqüències d'accions). Aquí podem veure els botons d'accions simples com són:

- Rebot ofensiu: quan un jugador atacant agafa el rebot en la cistella d'atac.
- Rebot defensiu: quan un jugador defensor agafa el rebot en la cistella de defensa.
- Recuperació: quan un jugador recupera la possessió de la pilota en una acció defensiva. Quan es crea una recuperació també s'ha d'anotar una pèrdua en l'equip rival (aquest pas no està contemplat)
- Tap: quan un jugador defensor bloqueja el tir d'un jugador atacant en la cistella de defensa.
- Pèrdua: quan un jugador perd la possessió de la pilota en una acció d'atac. Quan es crea una pèrdua, normalment també s'ha d'anotar una recuperació en l'equip rival (aquest pas no està contemplat).
- Falta: quan un jugador fa una falta, tant en una jugada ofensiva com defensiva. Només es polsa en el cas que sigui una falta de camp encara que l'equip defensor estigui en bonus, ja que per al 2+1 ja tenim un seqüència que s'adapta millor. El botó de tir lliure no està en aquest Fragment per espai però també hi hauria de ser per complementar les faltes amb l'equip en bonus.

Rebot O	Recupera	Perdua
Rebot D	Tap	Falta

Figura 53: Botons d'accions simples.

El prémer el botó de banqueta ens canvia la vista d'aquest Fragment de la pantalla i ens mostra tots els jugadors que hi ha disponibles. Marcant un jugador de banqueta i polsant el botó d'Acceptar tornarà a canviar la vista i haurem de seleccionar el jugador en pista que volem canviar per el

prèviament seleccionat en la banqueta. Passa el mateix si seleccionem 2 jugadors o més de la banqueta, que llavors haurem de seleccionar 2 o més jugadors de pista per fer la substitució.



Figura 54: Seqüència per a anotar les accions simples.

Cada botó de jugador està format per 2 paràmetres fixes i un de variable. Els fixes són el dorsal i el nom complet del jugador, i el variable són les faltes que duu el jugador en el transcurs del partit. La visualització directa de les faltes que duu cada jugador és interessant per a l'entrenador durant el transcurs d'un partit real.



Figura 55: Botó dels jugadors en pista.

### Marcador:

És la part més senzilla de l'Acta, ja que el marcador només consta d'un paràmetre estàtic, com són els noms del equips, i dos paràmetres dinàmics, que són el rellotge de compte enrere (amb els seus botons de Play i Pause) que es pot aturar cada cop que es pari el partit, i l'anotació general de cada equip. A més té un botó de guardar l'acta i un botó de sortida.



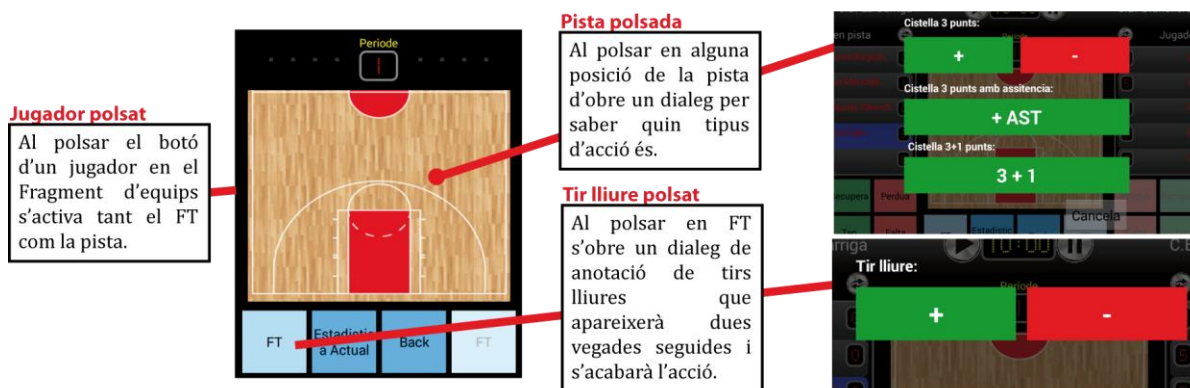
Figura 56: Fragment del marcador de l'acta.

### Pista:

La pista és un element una mica més complex, i la seva funcionalitat només és total quan hem marcat un jugador en el fragment d'equips. Els únics elements que sempre estan activats són el botó de Estadística, que

mostra l'estat actual del full d'estadística, i el de Back, que desfà l'acció anteriorment anotada.

Els següents elements només són activats quan s'ha marcat prèviament un jugador. La pista d'aquest Fragment conté una vista per a que es pugui detectar la posició en la zona de la pista, i a través d'una funció determinar si la posició és de 2 o 3 punts. A més conté el botó de tirs lliures (FT, que seria el següent pas d'una falta de camp), ja que no hi havia espai en el Fragment de equips.



**Figura 57:** Seqüència dels botons polsats en el fragment de pista.

També constà de 2 elements dinàmics com són el marcador de període, que ens marca el període del partit en el qual s'està jugant, i l'indicador de faltes d'equip per període.

### Controlador:

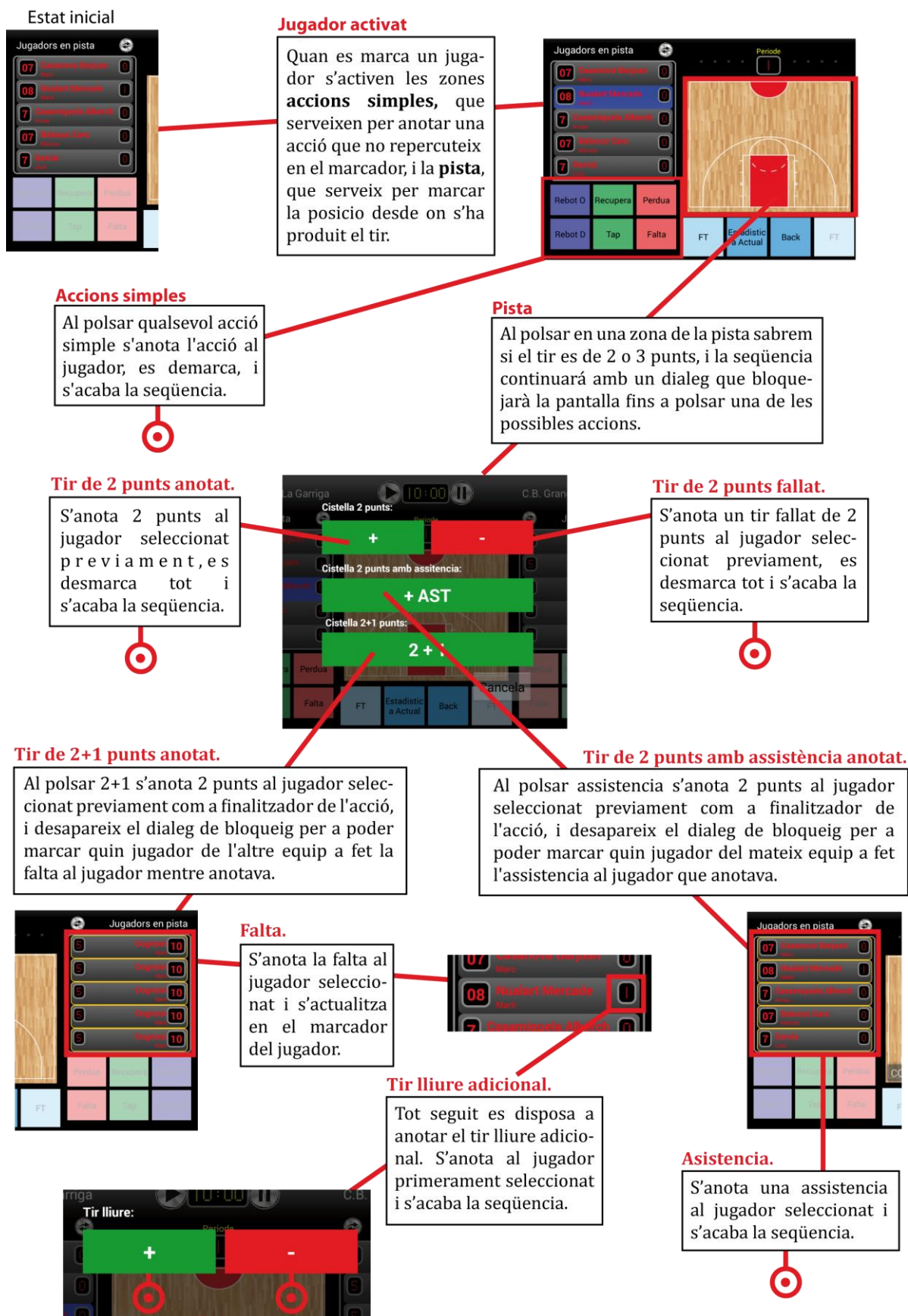
Arribem doncs a l'Activitat principal, que controlarà la comunicació entre pantalles i l'estat en tot moment de l'acció que volem anotar.

Per tant, veiem que el controlador no és una Activitat visual, sinó que està al darrere dels Fragments de la pantalla. És a dir, quan volem marcar una acció es visualitza en els fragments corresponents, però s'envia tota la informació al controlador, per avisar als altres Fragments que hi ha hagut un canvi, i si s'escau, actualitzar la part del fragment en cada cas.

Ara arriba el moment de veure quines seqüències de botons són necessàries per dur a terme les anotacions. L'experiència de l'usuari ha de ser intuïtiva per tal que no sigui complicat anotar una situació no trivial a l'acta. És a dir, no és el mateix anotar un tir de 2 punts d'un jugador, que anotar un tir de "dos més un" d'un jugador amb falta d'un altre jugador més el tir addicional. Per tant, la dinàmica d'anotació ha de començar sempre senyalant el jugador que fa l'acció, seguit de l'acció en sí, i depenent del tipus d'acció, activar els opcionals.



## Un exemple de seqüència quan polsem algun jugador:



**Figura 58:** Seqüència dels botons a pulsar per a dur a terme una acció complexa d'anotació.

## 5.2.6 Pantalla d'estadística

La següent pantalla és la d'Estadística. Aquesta també està formada per un conjunt de pantalles que faran més senzilla l'organització en quan a la visualització de les estadístiques.

La pantalla inicial d'estadística està formada per una pantalla canviant. Segons el botó que estigui premut veurem la disposició dels equips disponibles a consultar, o els jugadors, d'aquesta manera podrem consultar des de 2 nivells diferents. Nivell d'equips del club i nivell de jugadors del club.

Aquesta primera visualització constarà d'uns adaptadors d'informació en llistes, per poder visualitzar la informació que es vol en la llista a seleccionar.

Així doncs la disposició de la primera pantalla quedarà així:



Figura 59: Esquema original de la pantalla d'estadística per a seleccionar opció.

Seguidament, un cop premut un dels botons de qualsevol de les dues llistes, anirem a la següent pantalla. Aquesta ens mostrarà, en una llista semblant a la de la pantalla anterior, l'opció de seleccionar el partit del qual volem veure informació, l'equip o els jugadors.

Així que un exemple del que podríem trobar-nos en aquesta pantalla si premem un equip d'una categoria determinada i un jugador del mateix equip seria:



Figura 60: Esquema original de la pantalla d'estadística per a seleccionar partit.

D'aquesta manera tindrem una altra llista seleccionable en què, si premem algun dels seus botons, anirem a la pantalla final d'estadística, on tindrem una matriu on s'aniran col·locant totes les dades correctament.

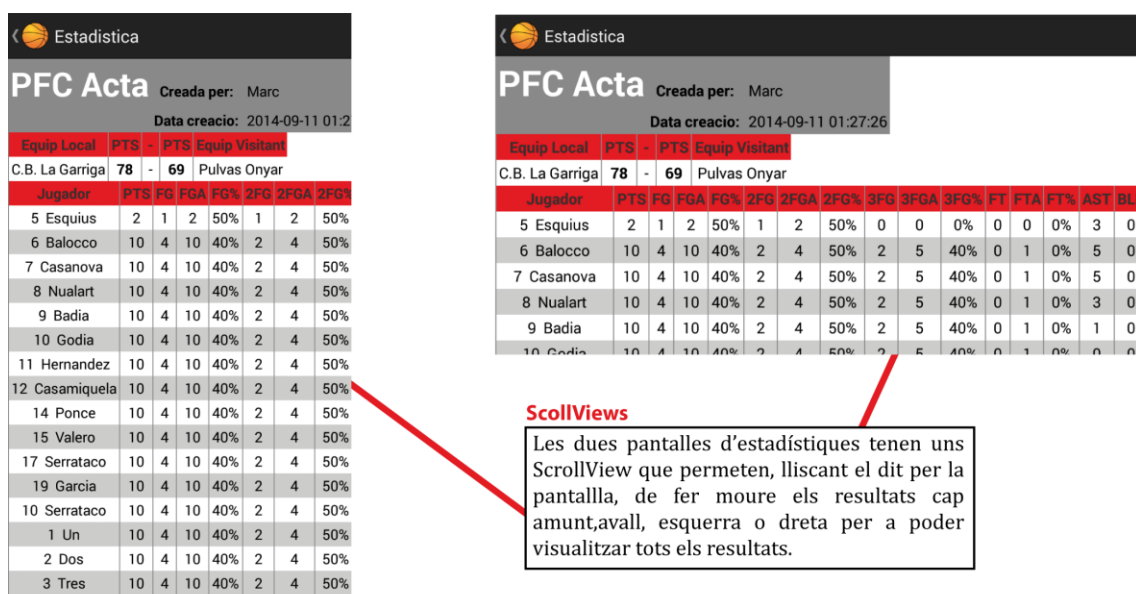


Figura 61: Esquema original de la pantalla d'estadística final.

## 5.2.7 Mòdul SQLite

A part de les diferents pantalles hi ha la classe de Gestor de base de dades que no s'ha explicat i queda una mica en segon pla, però que es un dels elements més importants de l'aplicació.

Aquest gestor anomenat *DatabaseHelper* conté totes les funcions per crear, actualitzar i eliminar la base de dades interna que forma part de l'aplicació. Dins d'aquesta classe hi haurà diferents funcions per gestionar totes les dades introduïdes.

El fet de tenir una base de dades interna agilitza molt la consulta d'informació, perquè no s'ha d'utilitzar la xarxa, però provoca un conflicte amb la integritat de les dades en el moment que el dispositiu no està connectat a la xarxa. Això es deu a que en moments que l'usuari no està connectat hi pot haver un actualització de la base de dades, i l'usuari al estar fora de la xarxa no poder rebre la notificació i la actualització pertinent.

Aquest problema no és tan greu ja que l'usuari pot fer l'acció en el moment en el qual rebí la notificació. Aquest problema és general de totes les aplicacions que requereixen estar connectats a la xarxa per dur a terme alguna acció. El fet de tenir connexió a Google Services per fer les notificacions garanteix que quan el dispositiu aconsegueixi estar connectat a la xarxa rebrà totes les notificacions endarrerides.

Més problemàtic és quan un usuari està fent una Acta sense connexió, ja que no pot enviar les notificacions a altres dispositius indicant que està fent una acta i quin és el seu estat. El problema s'agreuja quan aquest usuari acaba l'acta i no es pot actualitzar. Es podria enviar tota la informació per notificació, però com que hi ha tantes dades, seria una operació costosa a nivell de xarxa i s'ha optat per una solució manual. Les actes es poden enviar manualment des de l'apartat d'estadístiques, on tenim el llistat de totes les actes fetes.

## 5.2.8Mòdul Anàlisi

Aquest mòdul, ens facilita molt la feina alhora d'enviar les dades al servidor. JSON, és l'acrònim de JavaScript Object Notation, és un format lleuger per a l'intercanvi de dades. La simplicitat de JSON ha donat pas a la generalització del seu ús, i per tant, a utilitzar-se de forma massiva.

Amb aquest mòdul podem fer comandes GET, que són les comandes d'adquirir dades del servidor, per descarregar informació directament a la nostra aplicació. Fer servir JSON ens facilita el fet de desglossar les dades en el format que ens interessa per poder manipular-les correctament. Passa el mateix amb les comandes POST, que són les comandes encarregades de *parsejar*(agrupar) les dades que es manipulen en l'aplicació en el format correcte per ser enviades.

## 5.3 Unity

Ara toca començar a desenvolupar en Unity. Com hem seguit un ordre d'evolució en l'apartat de "disseny", seguirem el mateix en la implementació, ja que ens servirà per anar pas a pas.

Un cop polsem el botó de Tàctica, la nostra aplicació Unity s'executarà des de zero. El problema més gran que hi ha per a l'estructura de l'aplicació és que per a poder integrar totalment el nostre desenvolupament de Unity a Eclipse, hauríem de utilitzar la versió Unity Pro. Aquesta versió de Unity Pro només funciona sota llicència, cosa que fa més costós el desenvolupament, per tant, s'ha descartat l'opció d'utilitzar-lo.

La solució al problema és bàsica: generarem l'arxiu APK desde Unity i l'instal·larem juntament amb el paquet APK generat per Eclipse. Utilitzant una comanda especial, executarem l'APK de Unity des de la nostra aplicació, com si fos una altra aplicació independent.

Per motius de funcionalitat, també s'ha decidit descartar l'opció del simulador de defenses, ja que afegia molta complexitat alhora d'elaborar la solució final i la poca disponibilitat de temps ha requerit que finalment no s'acabés implementant.

Passa el mateix amb la visualització de jugades en 3D, que la seva complexitat i manca de temps no ha pogut acabar implementant-se, tot i que en la seva elaboració final s'ha pogut tocar una mica l'entorn en 3D i podem veure algunes imatges més endavant.

D'altra banda, s'ha dotat d'un joc senzill en 3D a l'aplicació, ja que les proves i tutorials que s'han dut a terme per desenvolupar en Unity ho feien possible. Com les figures del jugador de bàsquet són les mateixes que s'utilitzaven en l'intent de visualitzar jugada 3D, s'ha fet divertit el jugar amb els paràmetres per aconseguir muntar un joc entretingut.

### 5.3.1 Pantalla principal

La primera pantalla que ens trobarem serà la de Tàctica, que ha de constar de 3 botons. El botó d'editar jugada, el de visualitzar jugada, el de sortir (o enrere).

Funcionalment és molt senzilla, ja que només constarà de 3 botons que envien a una altra pantalla/nivell: El d'editor de jugades, el visualitzador de jugades i el de sortir.



**Figura 62:** Esquema original de la pantalla de menú principal.

### 5.3.2 Pantalla de l'editor

L'editor és la part més complexa d'aquest apartat de Unity, ja que és la que té més elements a controlar. Per a desenvolupar una jugada i després reproduir-la amb diferents peces movent-se en el tauler s'ha de guardar el moviment que fa el dit amb precisió. Hem de tenir en compte que hem d'estructurar la jugada en passos, ja que pot haver-hi més d'una fitxa movent-se en el mateix pas, per tant, hi haurà un marcadore amb el que controlem els passos per avançar en el desenvolupament de la jugada.

Per poder començar a muntar la jugada necessitem també botons per introduir diferents elements, com: fitxes blanques o negres, per representar els jugadors de cada equip a la pista, la fitxa que representa la pilota, botó per girar la pista, botons de control de passos, i lògicament botons d'esborrar i guardar la jugada.

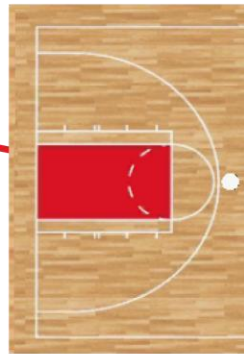
Tot això, funciona molt semblant a l'apartat d'Acta, ja que hi ha una classe principal que controla l'escena, amb el seu script definint cada funció. Els elements que anem introduint també tenen la seva classe definida, i podem controlar-los des de la classe principal.

Veiem aquí com queda definitivament la pantalla i una breu explicació dels passos que es segueixen per crear una jugada.

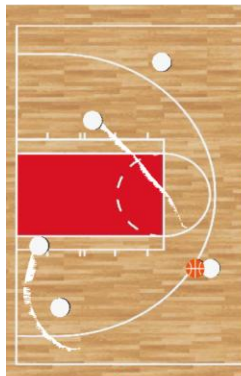
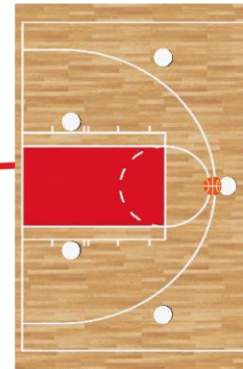




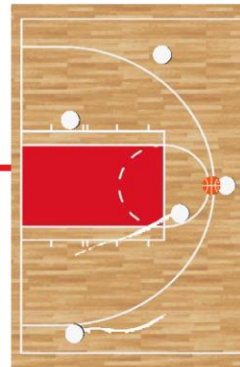
**Fitxa**  
Al iniciar l'editor, apremem la tecla Blanca i apareixerà una fitxa de color blanc a la pantalla.



**Fitxes**  
Si apremem el botó cinc vegades, les fitxes blanques apareixeran en la posició inicial més típica.



**Pas 1**  
Ara movem les fitxes que s'han de moure en el primer pas, marcant el seu moviment polsant-les i arrossegant-les per la pantalla. Un cop fet el moviment apremem el botó de Endavant per avançar un pas.



**Play**  
Un cop tenim la disposició de les fitxes correctament, apremem el botó de Play per començar a enregistrar la jugada.

**Pas 2**  
Fem el mateix que en el pas anterior, movem les fitxes que ens interessin i apremem Endavant.



**Salvar**  
Un cop tinguem tots els passos fets, apremem el botó de Salvar per emmagatzemar la jugada.



Figura 63: Seqüència per a enregistrar una jugada amb l'editor de jugades.

A cada pas emmagatzemem el moviment en una matriu, on a cada columna del vector guardarem la id de la fitxa, la posició inicial, la posició final, el pas en el que es produeix el moviment i un vector de mida normalitzada per emmagatzemar les posicions parcials en cada instant. Després d'això, en el moment de guardar la jugada, s'exporta en format xml i s'emmagatzema dins de la carpeta del programa. D'aquesta manera serà més senzilla la lectura de la jugada en els passos de reproducció, ja que hi ha funcions molt senzilles i pràctiques per llegir el format xml.

### 5.3.3 Pantalla de reproducció

Aquesta és més senzilla que l'anterior, ja que només s'ha d'interpretar el que hem guardat en el fitxer xml de la jugada. Per obtenir les dades en una matriu farem servir funcions de lectura de fitxers xml i desglossarem el seu contingut en un vector per cada pas.

Veiem que aquesta pantalla és similar a la d'edició, però únicament consta del controlador de passos. Aquí tindrem un botó de reproducció per simular la jugada en moviment, i controls de passos per avançar o retrocedir en la jugada per a la seva correcta comprensió.

Veiem les pantalles de la selecció de jugada i de reproducció:



**Figura 64:** Esquema original de la pantalla de reproducció.

### 5.3.4 Pantalla de reproducció 3D (no inclosa)

Tot i que la pantalla de reproducció en 3D no estigui inclosa, no vol dir que no se li hagi dedicat temps en la seva elaboració. La decisió de treure-la, a part de les hores que representava la seva implementació, va ser perquè vist d'una manera pràctica no s'assimila tan bé la jugada com en 2D. El fet de tenir peces complexes en moviment distreuen a l'usuari de l'essència del programa, que és l'assimilació d'una jugada.

Un altre aspecte en l'elaboració del 3D que feia que no fos possible implementar-la a curt temps era la complexitat dels moviments de les figures dels jugadors. El fet de no dominar el modelatge 3D en altres programes com Blender o 3D Studio Max obstaculitzaven l'elaboració de jugadors que es comportessin idòniament.





**Figura 65:** Esquema original de la pantalla de reproducció en 3D.

De totes maneres, el fet de treballar per intentar-ho ha donat certs resultats que dedicant-hi més temps podrien arribar a ser pràctics. Aquí podem veure unes captures de pantalla.

### 5.3.5Joc Encistella'm

El fet d'intentar desenvolupar en 3D ha desembocat en introduir un joc senzill i divertit de tirar a cistella amb un jugador. Els tutorials seguits per poder arribar a comprendre com funcionava Unity van fer que aquest mini-joc fos possible.

El joc consisteix en un jugador de bàsquet amb la pilota a les mans que tira a cistella des de diferents posicions del camp. La força del tir la decideix l'usuari polsant a qualsevol lloc de la pantalla, on el temps de pulsació representa la força amb la qual el jugador tirarà la pilota a cistella.



Figura 66: Esquema original de la pantalla del joc Encistella'm.

## 6. CONCLUSIONS

### 6.1 Conclusió

Un cop desenvolupada l'aplicació, arriba l'hora d'avaluar el treball. L'aplicació en sí té parts més complexes i parts més trivials, però en definitiva la valoració que en trec és molt positiva.

He acabat desenvolupant una aplicació funcional que compleix bona part dels objectius marcats inicialment. Hi ha parts que han sigut més complicades com la part de l'Acta digital. El fet de manipular tantes dades i poder fer tantes accions feien més complicada la manera d'introduir-les intuïtivament però crec que he trobat una solució correcta al problema.

Una altra part difícil de mostrar eren les estadístiques. Aquí, vist des del punt de vista del club, s'ha implementat correctament la manera que té l'usuari d'interactuar amb les dades. Aquesta part era un problema inicialment per la incertesa de com iniciar la cerca de les dades que es volien veure. El fet de dividir entre club i jugador ha provocat una manera més senzilla de fer les consultes a la base de dades, i en resum, la implantació de l'aplicació.

Una altra part que no s'introdueix en l'aplicació final es l'apartat 3D, tot i que en un inici era la prioritat, s'ha acabat aplicant el 2D per la seva facilitat de comprensió i agilitat visual. El factor més determinant a l'hora de no integrar-ho en l'aplicació és el temps de desenvolupament.

## 6.2 Valoració personal

Personalment valoro molt positivament l'experiència d'haver pogut desenvolupar una aplicació amb diferents tecnologies com són Eclipse, Unity i SQL Developer, cosa que ha fet incrementar els meus coneixements d'aquest sector per a poder-ho portar al món professional.

La presa de decisions per a afrontar els problemes de disseny eren complicades i tenien molta repercussió en l'estructura de l'aplicació en si, per tant, penso que s'han assolit bones solucions als problemes, encara que hi ha diferents aspectes que un cop acabat canviaria des de l'inici. Això sol passar quan l'experiència de l'usuari en diferents tecnologies és escassa, però sense tenir la sensació d'haver perdut el temps, ja que hi ha hagut molts dies que no tenia clara quina era la solució més adient en cada cas.

Per acabar, penso que s'ha treballat correctament i de manera constant, i això ha repercutit en un resultat final del tot satisfactori a nivell d'usuari.



## 6.3 Possibles millores

Durant el desenvolupament de l'aplicació s'han deixat de fer moltes coses que prèviament estaven establertes com a objectius ja que la disposició del temps a l'hora de desenvolupar-les era massa gran. Hi havia objectius que eren bastant complexes i per aquest motiu s'han aparcat en un segon terme per a que l'aplicació en sí estigues completa en quan a funcionalitat.

Llista de les possibles millores és la següent:

- ACTA: En els botons dels jugadors es podrien posar uns temporitzadors per a saber quan de temps porta el jugador a pista sense ser canviat per un company. Així podríem veure el desgast del mateix i avisar al entrenador del desgast del jugador.
- ACTA: Que les notificacions que indiquen que s'està fent una acta en comptes de únicament mostrar el resultat en la notificació, també mostrar l'estadística en el moment determinat del joc en el dispositiu que rep la notificació.
- ESTADÍSTICA: Fer una visualització més agradable amb l'usuari, perquè la taula d'estadístiques sigui més fàcil de llegir i manipular.
- SIMULADOR DE DEFENSES: El simulador de defenses era complicat de desenvolupar per la complexitat de les dades a manipular en un moment determinat, ja que la defensa s'hauria d'anticipar a certs moviments difícils de controlar. Però amb una bona pràctica d'aquest simulador es podria testear jugades fetes segons la defensa a atacar.

## 6.4 Problemes

El fet de treballar m'ha reduït el temps de desenvolupament, fet per el qual no s'han pogut implementar tots els mòduls que s'esperaven.

El problema més generalitzat que he tingut és la comprensió i l'aprenentatge del desenvolupament en l'editor Unity, ja que començant des de zero era complicat arribar a un resultat que em satisfés.

El més molest durant el desenvolupament en Eclipse era la constant compilació del codi en el dispositiu mòbil, ja que cada vegada que es testejava sempre fallava, i era feixuc corregir i tornar a executar la compilació, perquè tornés a fallar i així successivament fins a trobar la solució. El fet de no tenir la compilació en temps real com Unity era avorrit i a vegades estressant.

El problema que més m'ha dolgut és el fet que no podia accedir a la base de dades durant l'Agost, ja que el servidor Saturn de la UAB no estava operatiu.

## 7. REFERÈNCIES I BIBLIOGRAFIA

[Android Developers]

<http://developer.android.com/intl/es/sdk/index.html#download> - Últim accés: 15/04/2014. Pàgina de descàrrega de del paquet software que conté el producte Eclipse i l'Android SDK.

[Android Developers]

<http://developer.android.com/intl/es/reference/android/database/sqlite/SQLiteDatabase.html/> - Últim accés: 01/08/2014. Consulta del material de SQLite a Android Developers.

[Android Developers]

<https://developer.android.com/intl/es/google/gcm/index.html>- Últim accés: 15/08/2014. Consulta del material GCM a Android Developers.

[Google Developers Console]

<https://console.developers.google.com/project/apps~pfc-bandouck-01/apiui/credential?authuser=0download> - Últim accés: 01/09/2014. Interfície web per a la gestió, vista del tràfic i autenticacions relacionades amb el projecte.

[StackOverflow]

<http://stackoverflow.com/questions/> - Últim accés: 17/08/2014. És una pàgina de discurs per a consultar preguntes que fa la gent i com solucionar certs aspectes de la implementació.

[Android Hive]

<http://www.androidhive.info/>- Últim accés: 18/08/2014. Pàgina de tutorials per a programació en Android.

[Tutorials Point]

[http://www.tutorialspoint.com/android/android\\_network\\_connection.htm](http://www.tutorialspoint.com/android/android_network_connection.htm)- Últim accés: 10/08/2014. Pàgina de tutorials per a programació en Android.

[Unity]

<http://unity3d.com/> - Últim accés: 21/05/2014. Pàgina principal del producte Unity.

[PHP]

<http://php.net/> - Últim accés: 04/08/2014. Es la pàgina oficial del llenguatge php.