

Memòria PFC

Programació d'aplicacions per a mòbils usant HTML5

Valoració de parells paraula-imatge per a un estudi psicològic

Pau Roura Brun
Enginyeria Informàtica

1. Definició del Projecte	3
1.1 Descripció	3
1.2 Objectius	4
2. Metodologia	5
2.1 Terminis	5
2.2 Calendari del projecte	6
3. Tecnologies usades	8
4. Anàlisi i Disseny	9
4.1 Histories D'usuari	9
4.2 Diagrama de Classes	12
4.3 Disseny d'interfície.....	14
5. Implementació.....	17
6. Possibles millores del projecte	22

1. Definició del Projecte

1.1 Descripció

Aquest es un projecte que comença a instàncies del departament de Psicologia de la URV, per la necessitat d'una aplicació per portar a terme un experiment, com a part d'un estudi en l'àrea de Psicologia del Llenguatge. En aquest experiment, es vol analitzar com associen les persones una sèrie d'imatges amb conceptes relacionats.

Per aquest motiu, es crea una aplicació per a mòbils que permeti als participants en l'experiment realitzar una sèrie de tests còmodament en qualsevol moment i que transmeti les dades als responsables de l'experiment. L'aplicació demanarà a l'usuari que es registri a partir del correu, per poder tenir l'usuari identificat, i es connectarà a la nostra base de dades per guardar les dades del participant i per descarregar alguns tests per que l'usuari pugui realitzar en qualsevol moment, inclús quan no tingui connexió a la xarxa.

Un cop registrat, l'usuari podrà, en qualsevol moment, iniciar un test. Abans d'iniciar el test en si, se li presentarà una pagina amb les instruccions sobre com realitzar la prova. En aquest punt, tindrem dos tipus de tests:

En el primer tipus, se li presentarà una pantalla amb una imatge i una paraula, en el que l'usuari haurà de valorar, en una escala de l'1 al 7, el grau de relació que creu que hi ha entre elles.

En el segon tipus de test, se l'hi presentarà primer la imatge sola, per que l'usuari es formi una idea sobre el que representa per si mateix, i al cap d'uns segons se li presentarà la paraula relacionada i se li demanarà el grau de relació, també de l'1 al 7, tenint en compte la idea que tenia en ment.

En ambdós casos ens interessarà la puntuació que ens ha donat l'usuari, i el temps que ha tardat en donar una resposta.

1.2 Objectius

L'objectiu del PFC es desenvolupar una aplicació que ens permeti dur a terme el nostre estudi sobre un gran nombre de participants. Això es el que ens fa decidir per una aplicació mòbil, que permeti a l'usuari participar en l'experiment en qualsevol moment i de la manera mes còmode

D'aquí també traiem dos requisits importants:

- Primer, que per l'usuari sigui el mes fàcil possible realitzar l'alta en el projecte, i iniciar la seva participació de seguida
- Segon, en la mateixa línia, que l'aplicació permeti a l'usuari realitzar tests offline en qualsevol moment.

Com que l'usuari participa de forma voluntària en l'experiment i no obté cap benefici directe d'usar la nostra aplicació, som conscients que qualsevol obstacle o imprevist pot portar a l'usuari a posposar i/o abandonar la seva participació.

Per això l'alta d'usuari la farem el mes simple possible: li demanarem les dades significatives per l'estudi i un correu de contacte, però no li demanarem que confirmi la seva adreça ni muntarem tot un sistema d'alta i gestió d'usuari, ja que no tenim realment cap resposta que li vulguem donar mitjançant l'aplicació.

2. Metodologia

El desenvolupament del projecte es condiona al calendari de l'assignatura, estructurat amb 3 entregues parcials i dues finals. La primera entrega correspon a la descripció i planificació de la proposta, mentre que la resta corresponen al desenvolupament pròpiament dit.

2.1 Terminis

Octubre

- 09/10 – PAC1
 - Concreció de la proposta.
 - Pla de projecte. Objectius i calendari
- 30/10 – PAC2
 - OA1 – aprendre a usar parse.com – 1 setmana
 - OA2 – projecte PhoneGap – 1 setmana
 - H2 – mostrar en pantalla paraula-imatge – 1 setmana

Novembre

- 27/11 – PAC3
 - H3, OA4 – Puntuació de l'usuari, obtenir temps emprat i guardar a la BD – 1,5 setmanes
 - H4 – Alta d'usuari – 1 setmana
 - H5, OA3 – test complet – 1,5 setmanes

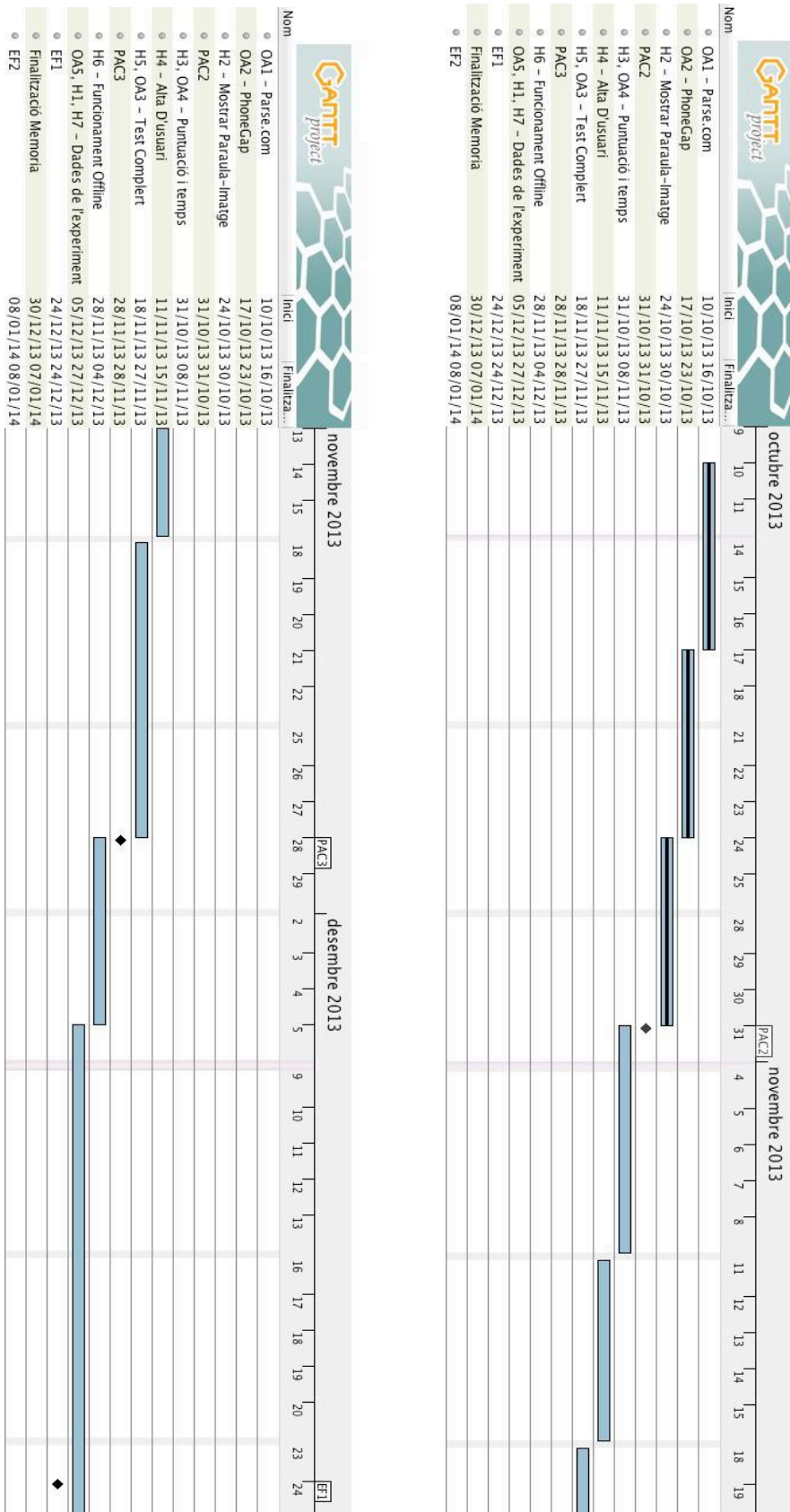
Desembre

- 23/12 – Entrega Final (1)
 - H6 – funcionalitat offline -1 setmana
 - OA5, H1, H7 – Carrega i recuperació de dades, amb interfícies -3 setmanes

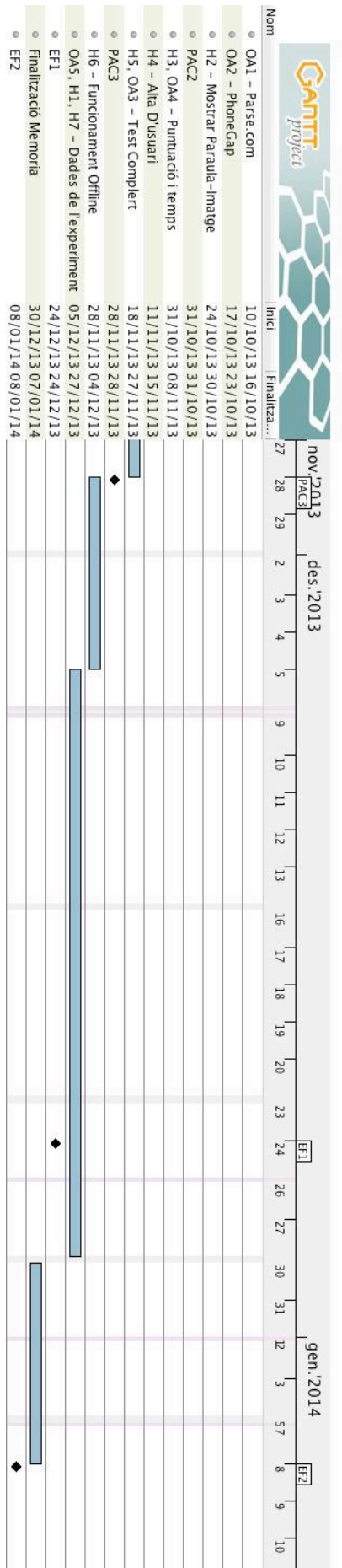
Gener

- 08/01 – Entrega Final (2)
 - OA5, H1, H7 – Carrega i recuperació de dades, amb interfícies -3 setmanes

2.2 Calendari del projecte



Pau Roura – PFC - Programació d'aplicacions per a mòbils usant HTML5
 Memòria



3. Tecnologies usades

Donat que volem arribar al màxim possible de participants ens interessa tenir una aplicació multiplataforma que funcioni tant en entorns Android com iOS, per això crearem la interfície en HTML5 i JavaScript, donat que són tecnologies que tots els sistemes mòbils suporten. L'aplicació inicial es realitzarà per iOS amb la vista posada a una possible ampliació de SO suportats.

Per això, per la interfície farem servir **PhoneGap**, un framework que ens permet crear les interfícies en aquestes tecnologies, i ens ofereix una sèrie de APIs, per accedir d'una manera estandarditzada a les funcionalitats que ofereixen els dispositius mòbils.

De cara a les interfícies, també usarem **jQuery Mobile**, una llibreria de JavaScript molt popular que simplifica l'accés i modificació dels documents HTML.

Per la gestió de les dades, usarem **Parse.com**, que es una eina per gestionar dades basada en cloud, que es perfecte per la nostra aplicació, que necessita reunir les respostes de tots els participants. A més, Parse treballa sobre els propis servidors de l'empresa responsable, fet que ens estalviarà haver de gestionar-los nosaltres mateixos

Parse ens permetrà tenir les dades de tots els tests al cloud, i també ens ofereix eines per descarregar dades al dispositiu on està instal·lada la nostra aplicació. D'aquesta manera, l'aplicació es descarregarà alguns tests preparats per que l'usuari els pugui realitzar en qualsevol moment i poder pujar les respostes al cloud el proper cop que tingui connexió.

La carrega i recuperació de les dades de l'experiment les farem en format **JSON**, que es un format estàndard, derivat de JavaScript, i que nombrosos programes i llenguatges son capaços de generar i interpretar.

Finalment usarem **WebSQL** per guardar les dades tant dels tests a realitzar com dels que ja ha contestat l'usuari i estan pendents de trametre als nostres servidors.

Usant aquestes tecnologies podrém desenvolupar l'aplicació usant un patró model-vista-controlador, amb les vistes en HTML5 (PhoneGap), els controladors en JavaScript (PhoneGap+jQuery) i el Model en Parse.com

4. Anàlisi i Disseny

4.1 Histories D'usuari

H1	Carrega de les dades de l'estudi
Descripció Com a responsable de l'estudi, vull poder carregar les paraules i imatges relacionades sobre les que vull fer l'estudi. Això implica tenir el model de dades creat al núvol i una pagina d'accés per al responsable on pugui pujar les dades. Inicialment les dades les pujarem directament serialitzades en format JSON.	

H2	Presentació d'una pregunta a l'usuari
Descripció Com a desenvolupador, vull poder mostrar en pantalla una parella paraula- imatge. Això comporta crear la interfície per mostrar-les a l'usuari i recuperar les dades del servidor.	

H3	Resposta d'usuari
Descripció Com a usuari, vull poder puntuar el grau de relació que crec que hi ha entre la paraula i la imatge que se m'ha mostrat. S'haurà de guardar la puntuació que l'usuari ha donat a la BD, i calcular el temps que ha tardat en donar la resposta.	

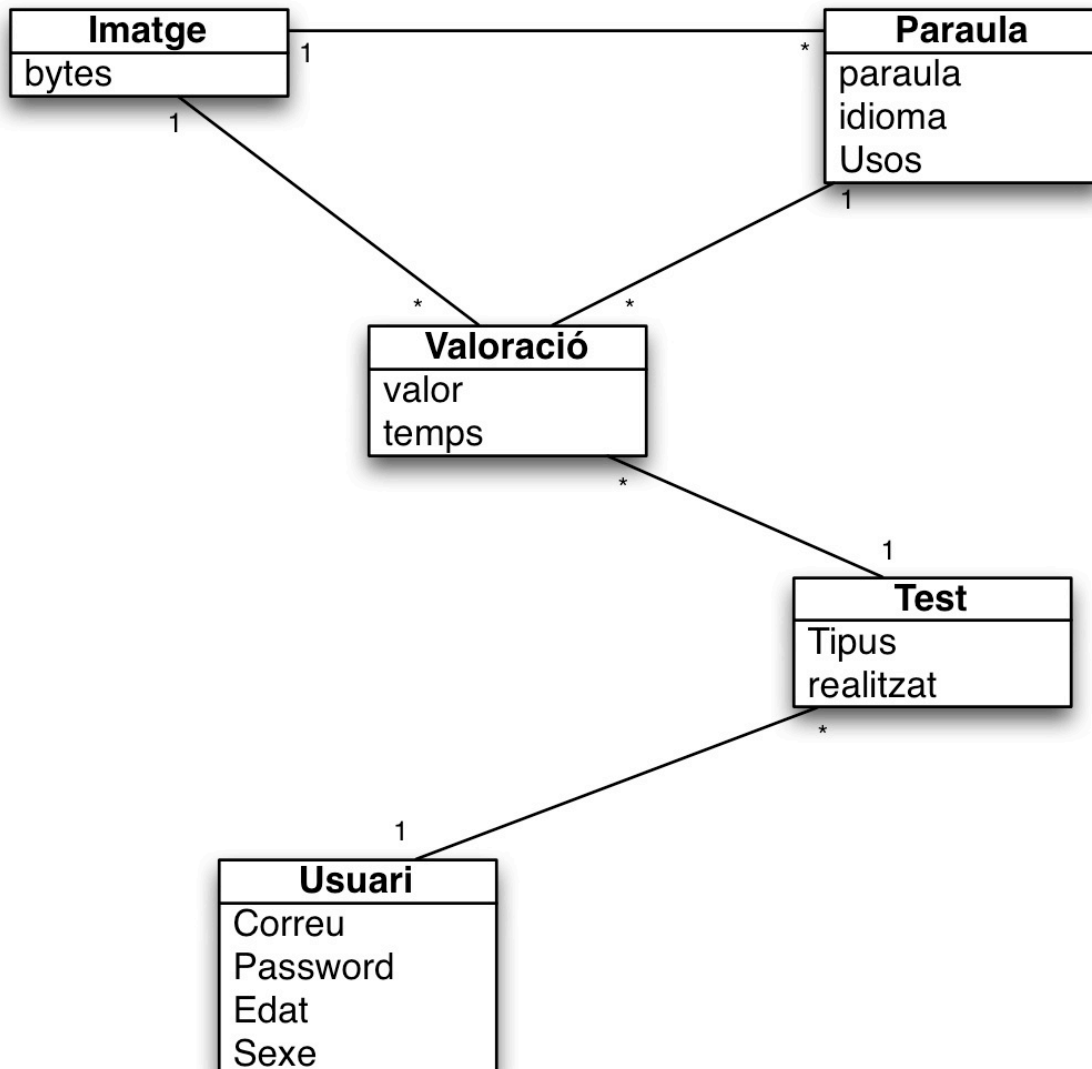
H4	Alta d'usuari
Descripció L'usuari es dona d'alta com a participant en l'experiment. A l'usuari se li demanaran dades que l'identifiquin, com pot ser el correu electrònic, i dades rellevant per l'experiment, com poden ser edat i sexe.	

H5	Realització d'un test complert
Descripció Com a usuari, realitzar un test complert. Això implica que l'usuari pugui seleccionar un dels tests preparats i que l'aplicació seleccioni, dins d'aquell test, quines parelles paraula-imatge li mostrarem, tenint en compte criteris d'aleatorietat, no repetició de paraules, i que, entre els diferents usuaris, volem tenir valors per totes les parelles possibles.	

H6	Possibilitat de realitzar tests offline
Descripció Com a usuari vull poder realitzar un test complert en qualsevol moment, independentment de la connectivitat que tingui el meu dispositiu. L'aplicació, per tant, haurà de ser capaç de guardar en local alguns tests preparats per i de transmetre els resultats quan recuperi la connectivitat amb el servidor.	

H7	Recuperació dels resultats de l'estudi
Descripció Com a responsable de l'estudi, vull poder recuperar els resultats de l'estudi. Tindrem una pagina d'accés on podrem accedir a un fitxer JSON amb tots els resultats de l'experiment.	

4.2 Diagrama de Classes



Imatge es la classe que emmagatzema la imatge en binari.

Paraula conté els diferents conceptes que volem relacionar amb les diferents imatges. A part, identificarem l'idioma de la paraula per saber si hi ha diferències, i el número de cops que s'ha usat aquella paraula en el total de tests, per intentar efectuar un nombre similar de proves amb totes les paraules. Una imatge pot tenir diverses paraules relacionades.

Valoració correspon al valor que ens ha donat un usuari per aquella paraula i concepte. Tot i que en la majoria de casos la **Imatge** presentada a l'usuari serà la

corresponent a la **Paraula** a la que enllaça valoració, en el test presentarem també parells no relacionats per observar les respostes dels usuaris.

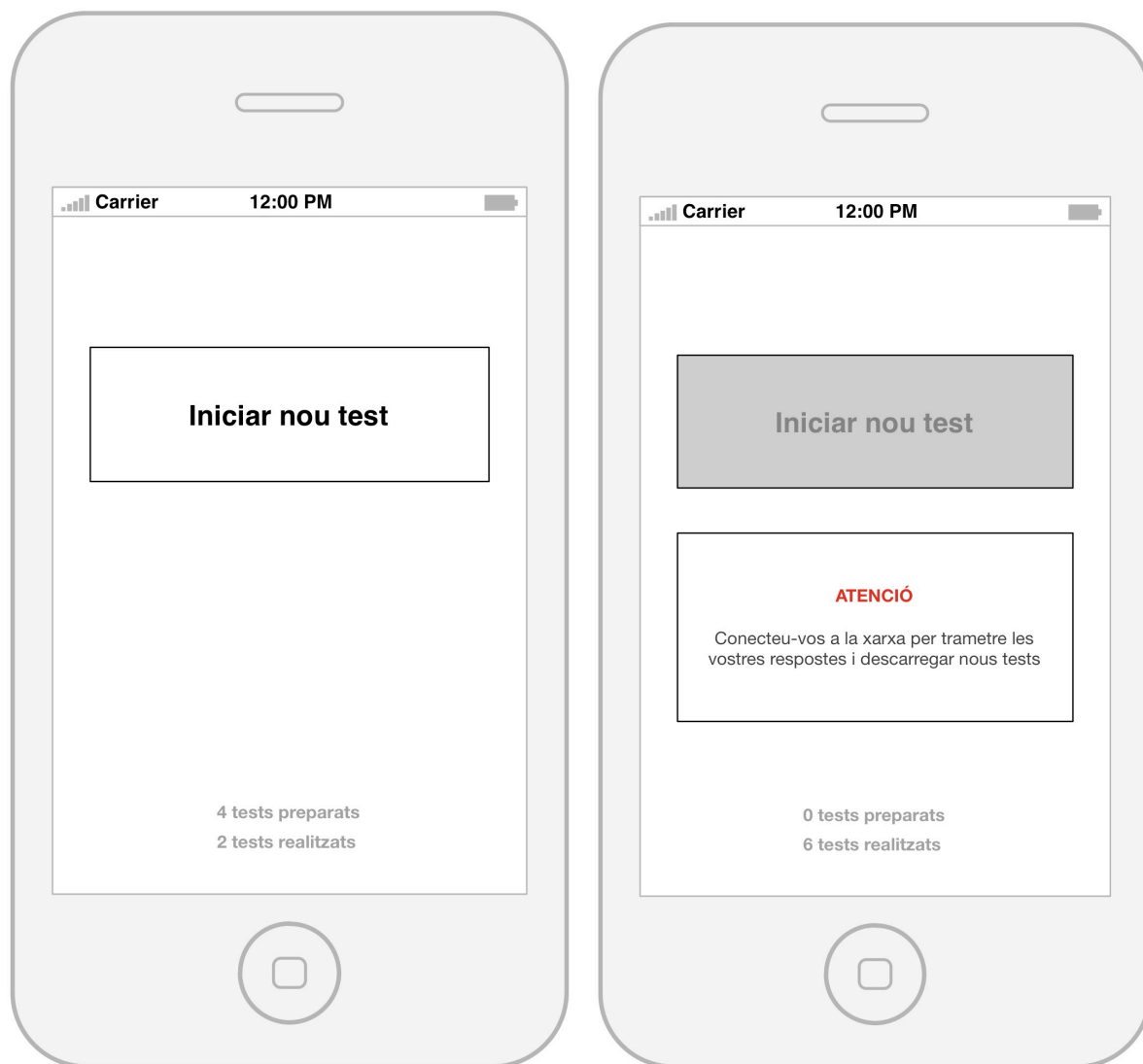
Un **Test** es un conjunt de valoracions, voldrem saber de quin tipus es (directe Imatge-Paraula o en dues fases), i si ha estat completat (per que primer preparem els tests per que l'usuari els pugui fer offline. Els Test seran únics per cada usuari i generats aleatòriament.

A mes, guardarem les dades de l'**Usuari** que ens puguin interessar.

4.3 Disseny d'interfície



El primer cop que l'usuari inici l'aplicació, se li mostra una pantalla amb el botó d'iniciar test present (per que l'usuari tingui clar que està al lloc correcte pels tests) però desactivat i el d'alta d'usuari, que serà l'única opció disponible. Aquest botó el portarà a la pantalla d'alta d'usuari, on li demanarem les dades que ens interessen. Com hem comentat, l'alta es el mes simple possible, l'usuari no te cap opció de gestió ni d'accés a les seves dades i resultats, així que no es necessari ni tant sols un mot de pas per identificar-se posteriorment. Només ens interessen les dades de l'usuari que respondrà tests en aquest dispositiu.

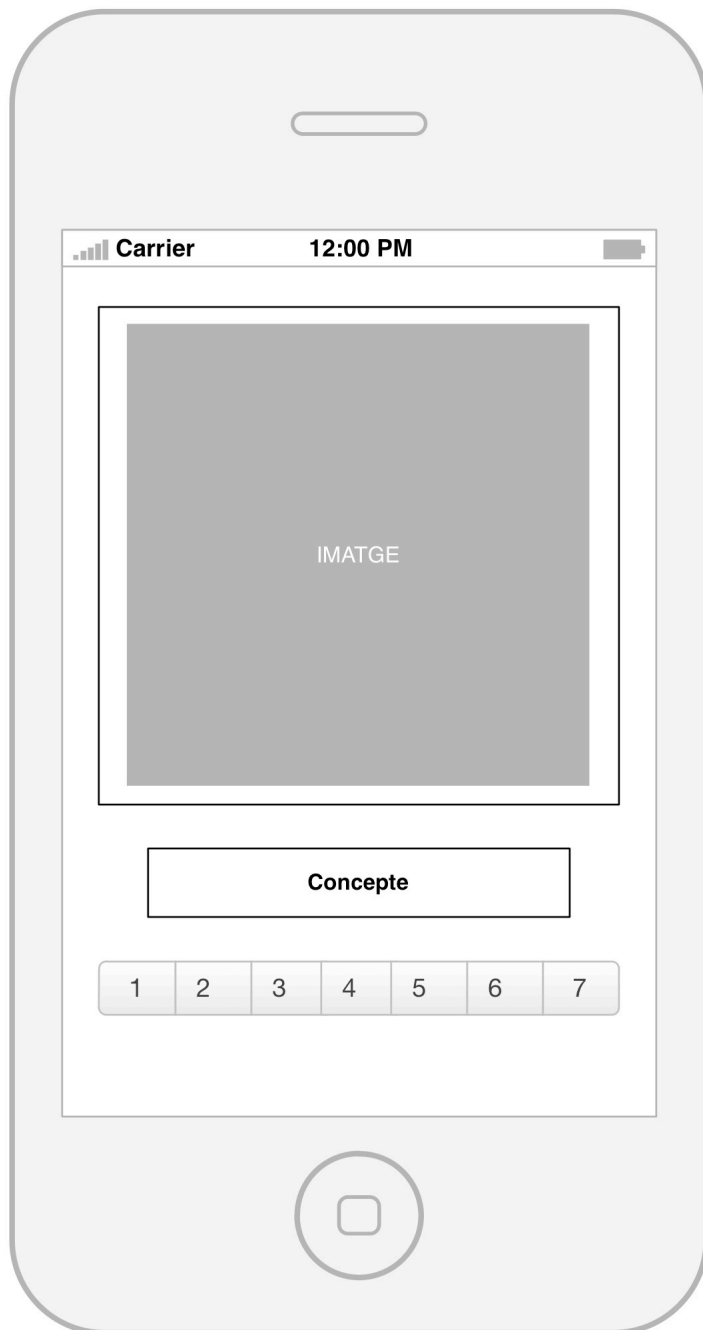


Un cop donat d'alta, la pantalla principal que veu l'usuari es força senzilla, amb la opció d'iniciar un nou test, i amb una petita indicació dels tests que té preparats (disponibles per fer en local. Se'n baixaran de nous quan la connexió estigui disponible) i realitzats (els tests que tenen pendent d'enviar les dades al servidor).

Si l'usuari ja ha realitzat tots els tests preparats, al iniciar l'aplicació sense connexió la opció d'iniciar un nou test apareixerà desactivada (per que l'usuari vegi que està a la pagina correcte, però la opció no està disponible) i una indicació del motiu pel que no pot iniciar els tests.

Quan iniciï el test, l'usuari veurà una pantalla amb un text explicatiu del tipus de test que realitzarà (directe o per fases)

El test en si consistirà d'una sèrie de pantalles amb la imatge i el concepte que volem puntuar, així com una gradació de l'1 al 7 preparada. Si un dels paràmetres que volem controlar es el temps que l'usuari tarda en respondre, es el mes adient per que l'usuari passi el menor temps possible en la interfície.



5. Implementació

5.1 Alta d'usuari

La primera interacció amb l'usuari serà l'alta d'usuari. Donat que no necessitem un login real, sinó que l'usuari s'identifiqui un únic cop, aprofitarem el localStorage que ofereix l'estàndard HTML5 per guardar l'ID d'usuari. Només si no tenim l'id d'usuari establert mostrarem l'enllaç a la pantalla d'alta

```
var local = window.localStorage;

var userId = local.getItem("UserID");

if (userId)
{
    document.getElementById('usrLink').style.display = 'none';
} else {
    document.getElementById('usrLink').style.display = 'block';
}
```

5.1 Creació dels Tests

El següent pas es la creació del test que volem que l'usuari realitzi. Donat que els test son únics per part de l'usuari, la creació (selecció dels parells paraula-imatge que componen aquell test) la farà la pròpia aplicació. L'aplicació crearà els objectes Test i Valoració al servidor. El tipus de test es determina aleatòriament

```
var newTest = new Parse.Object("Test");
newTest.set("usuari", userId);
var type = (1+Math.floor(Math.random()*2))+'';
newTest.set("Tipus", type);
newTest.set("Realitzat", false);

//Creem un nou test
newTest.save().then(function(newTest)
```

El test i les valoracions creats es guardaran en local a una base de dades WebSQL per que estiguin disponibles offline. Tot i que Parse.com te un sistema de cache, es un sistema automàtic que guarda les ultimes peticions realitzades. Ja que no tenim cap control sobre les dades que s'han guardat en local, no ens serveix pels nostres propòsits.

Com hem dit, la selecció dels parells paraula-imatge els fa l'aplicació. La selecció de les paraules es fa seleccionant les paraules menys usades en el global de tests. Per això a nivell de servidor guardem el nombre de cops que s'ha usat en els tests del conjunt d'usuaris.

```
var paraula = Parse.Object.extend("Paraula");
var query = new Parse.Query(paraula);
query.limit(TEST_ITEMS);
query.ascending("Usos");

query.find({
```

Finalment, els tests contenen una sèrie de parells “incorrectes” escollits aleatòriament. A l'aplicació se li configura la proporció d'items aleatoris que tindrà i ignorarà la relació paraula-imatge que li indica al servidor i escollirà una imatge a l'atzar.

```
var rand_items = results.length * RANDOM_ITEMS_PCT;
```

```
var img = Parse.Object.extend("Imatge");
var queryImg = new Parse.Query(img);

if(random)
{
    queryImg.count({
        success: function(count)
        {
            console.log('novaValoracio random Img count:'+count);
            var skp = Math.floor(count * Math.random());
            queryImg.skip(skp);
            retrieveImage(queryImg, newVal, paraula, deferred);
        },
        error: function(error)
        {
```

```
        deferred.resolve();
    }
});
}
else
{
    console.log('novaValoracio ImgId');
    queryImg.equalTo("objectId", imgId);
    retrieveImage(queryImg, newVal, paraula, deferred);
}
```

La imatge es guarda a la BD local, en base64. Això ens permetrà també insertarla directament a la pagina de puntuació.

5.3 Realització del test

Tot seguit, es selecciona un dels tests que tenim guardats en local i s'activa el link d'inici de test. Aquest ens passa a la pantalla de puntuació, on es selecciona una de les valoracions que tenim emmagatzemades. La selecció es també aleatòria, per evitar casos com que l'ordre entre paraules es mantingui en tests diferents o que els parells incorrectes apareguin sempre en les mateixes posicions.

```
tx.executeSql('select * from valoracio where test = ? and valor is null;', [testId],
function(tx, results){
    if(results.rows.length===0)
    {
        console.log('loadImage cap valoració');
        window.location="./index.html";
        return;
    }
    var i= Math.floor(Math.random()*results.rows.length);
    var row = results.rows.item(i);
```

Com veiem, si no trobem cap valoració sense valor establert es senyal que hem acabat el test i tornarem a la pagina principal. Per obtenir el temps que tarda l'usuari a donar la resposta el `Date().getTime()` de Javascript ens dona el temps en ms, precisió suficient per les nostres necessitats. Si el test es del tipus 2, per parts, no iniciarem el timer de resposta fins que haguem amagat el panell d'imatge i mostrat el de valoració.

5.4 Enviament de respostes

La resposta que ens dona l'usuari la guardarem a la BD local i transmetrem totes les respostes del test de cop. Com veiem cap dels passos de la puntuació necessita connexió a la xarxa.

Al carregar la pagina principal, aquesta comprova si hi ha dades pendents de trametre cap als servidors, comprovant si algun dels tests presents a la BD te respostes a totes les valoracions que en formen part. Això ens servirà tant per els tests que acabem de completar com per els que realitzem offline un cop recuperem la connectivitat.

La combinació d'aquests dos punts també permetrà que l'usuari deixi tests a mitges (tot i que no li donem l'opció, pot tancar l'aplicació) i completi les valoracions pendents a posteriori.

5.5 Carrega de les dades de l'estudi

Les persones que duguin a terme l'experiment, han de poder carregar les dades de l'experiment, els parells paraula-imatge que es presentaran, en bloc. Per això els oferim la possibilitat de carregar les dades en format JSON,. El JSON que requerirem serà un array d'objectes que contindran dos camps, amb la paraula i la imatge en base64.

```
[{  
  "nom" : "Paraula" ,  
  "imatge" : "Base64"  
}, {  
  "nom" : "Paraula" ,  
  "imatge" : "Base64"  
}]
```

5.6 Recuperació de resultats

Igualment, els responsables de l'experiment podran recuperar les respostes que han donat els participants, també en un arxiu en format JSON. En aquest arxiu tindrem, per cada usuari, els tests que ha realitzat i, dins d'aquests tests, les paraules que els havíem presentat, la imatge que havien vist en cas que no fos la corresponent a la paraula, i la puntuació que ens han donat.

```
[{
  "usuari":"IDUsuari",
  "nom":"NomUsuari",
  "sexe":"Home/Dona",
  "edat": edat,
  "correu":"correu ",
  "tests":[
    {"test":"IDTest", "valoracions":[
      {"paraula":"Paraula", "valor":0-7},
      {"paraula":"Paraula", "valor":0-7},
      {"paraula":"Paraula", "imatge":"Paraula", "valor":0-7},
      ...
      {"paraula":"Paraula", "valor":0-7}]
    },{"test":"IDTest", "valoracions":[
      {"paraula":"Paraula", "valor":0-7},
      ...
      {"paraula":"Paraula", "valor":0-7}]
    }
  ]
}],{
  "usuari":"IDUsuari",
  ...
}]
```

6. Possibles millores del projecte

Versió per sistemes operatius Android

De cara a arribar al màxim de persones possibles, podem d'aprofitar les tecnologies que hem usat per la implementació del nostre projecte, que son estàndards multi plataforma (Html, Javascript) per intentar ampliar al màxim el nostre públic potencial.

Suport per a diversos idiomes

Una altra possible ampliació que ens proposen seria la possibilitat de poder realitzar els tests en diferents idiomes. Per això, les paraules estarien associades a l'idioma en el qual estan escrites. Així mateix, durant l'alta l'usuari indicaria quins idiomes coneix. Durant la creació del test, es seleccionaria un d'aquests idiomes a l'atzar i el test estaria format íntegrament per paraules en aquell idioma.