

FentPais



TFC- Aplicacions de dispositius mòbils –ANDROID

Alumne: Raül Muñoz

Consultor: Albert Grau

Resum

FentPais és una cooperativa que ofereix als seus clients la possibilitat de regalar unes targetes per fer certes activitats.

La Aplicació vol ajudar a la cooperativa a que els clients puguin cercar les activitats que tenen més a prop i que puguin compartir les seves experiències amb altres usuaris.

Paraules clau:

FentPais, Java, Android,

*M'agradaria dedicar aquest treball a totes les persones que m'han ajudat,
especialment a la meva família pel seu suport incondicional.*

Índex

| | |
|---|---------|
| 1. Introducció | Pag. 8 |
| 1.1 Característiques tècniques | Pag. 8 |
| 1.1.1 De software..... | Pag. 8 |
| 1.1.2 De hardware | Pag. 9 |
| 1.2 Abast global del projecte | Pag. 9 |
| 1.3 Resultats esperats | Pag. 10 |
| 1.4 Planificació..... | Pag. 11 |
| 1.4.1 Planificació inicial..... | Pag. 11 |
| 1.4.2 Planificació final | Pag. 11 |
| 2. Funcionalitats..... | Pag. 12 |
| 2.1 Característiques funcionals | Pag. 12 |
| 3. Disseny centrat en l'usuari..... | Pag. 13 |
| 3.1 Usuari i context d'ús | Pag. 13 |
| 3.2 Investigació: Contextual Enquiry | Pag. 14 |
| 3.3 Perfils d'usuari | Pag. 16 |
| 3.4 Context d'ús..... | Pag. 17 |
| 3.5 Anàlisi de tasques..... | Pag. 17 |
| 3.6 Escenari d'ús | Pag. 19 |
| 3.7 Prototips horitzontals d'alta fidelitat | Pag. 21 |
| 4. Arquitectura de l'aplicació..... | Pag. 27 |
| 4.1 Classes del sistema..... | Pag. 27 |
| 4.1.1Classe PrincipalActivity..... | Pag. 28 |
| 4.1.2 Classe Validar..... | Pag.29 |
| 4.1.3 Classe LlsitaComentaris | Pag. 30 |

| | |
|---|---------|
| 4.1.4 Classe Mapa..... | Pag. 31 |
| 4.1.5 Classe LlistaEstabliments..... | Pag. 32 |
| 4.1.6 Classe ListAllotjaments..... | Pag. 33 |
| 4.1.7 Classe ListCultura..... | Pag. 34 |
| 4.1.8 Classe ListEnoturisme | Pag. 35 |
| 4.1.9 Classe ListGastronomia | Pag. 36 |
| 4.1.10 Classe ListNatura | Pag. 37 |
| 4.1.11 Classe Usuari | Pag. 38 |
| 4.1.12 Classe ComentarisAdapter..... | Pag. 39 |
| 4.1.13 Classe Comentaris | Pag. 40 |
| 4.1.14 Classe EstablimentsAdapter | Pag. 41 |
| 4.1.15 Classe Establiments | Pag. 42 |
| 5. Disseny interfície gràfica..... | Pag. 43 |
| 5.1 FentPais..... | Pag. 43 |
| 5.2 Validar usuari..... | Pag. 45 |
| 5.3 Pantalla comentaris | Pag. 46 |
| 5.4 Pantalla a prop meu..... | Pag. 47 |
| 5.5 Pantalla establiments..... | Pag. 48 |
| 5.6 Pantalla llista allotjament | Pag. 49 |
| 5.7 Pantalla llista gastronomia..... | Pag. 50 |
| 5.8 Pantalla llista enoturisme..... | Pag. 51 |
| 5.9 Pantalla llista cultura..... | Pag. 52 |
| 5.10 Pantalla llista natura | Pag. 53 |
| 5.11 Pantalla usuari | Pag. 54 |
| 6. Avaluació de la aplicació | Pag. 55 |

| | |
|--|---------|
| 7. Variacions respecte a la proposta inicial | Pag. 56 |
| 8. Línies futures de treball | Pag. 57 |
| 9. Bibliografia | Pag. 58 |

Introducció

Avui en dia, molta gent disposa de un dispositiu mòbil Android, és per això que la cooperativa FentPais ha volgut fer una aplicació pels seus socis.

Aquesta aplicació permetrà compartir les experiències dels clients un cop hagin realitzat alguna activitat que ofereix la cooperativa. Els socis també podran veure quines activitats s'estan realitzant a prop de la seva ubicació. I per finalitzar, podran veure una descripció de les activitats que ofereix la cooperativa.

En un principi la idea era que la aplicació es comuniqués amb la base de dades del servidor i actualitzés les dades, però per falta d'enteniment entre les dues parts això no s'ha pogut implementar. Així que la aplicació només mostra les dades que guardades.

1.1 Característiques tècniques

1.1.1 De software

Per a desenvolupar aquesta aplicació s'ha utilitzat l'IDE Eclipse.

Les proves s'han realitzat amb dispositiu amb Android 4.4.2, tot i que la aplicació funciona amb com a mínim un Sistema operatiu Android 2.2

1.1.2 De hardware

L'ide eclipse estava instal·lat en un Windows 7 de 64 bits. El ordinador personal era un intel i5 amb 8 Gb de RAM.

L'aplicació es provava en un Nexus 5.

1.2 Abast global del projecte.

L'abast global del projecte engloba l'anàlisi i el disseny e implementació de l'aplicació.

Es divideix en vaires fases: planificació, anàlisi, desenvolupament i documentació i presentació.

La fase de planificació finalitzarà quan s'obtinguin un plantejament general del projecte i unes dades estimades per a les diferents fases i subfases.

La fase d'anàlisi finalitzarà quan s'obtingui una sèrie de requeriments tant funcionals com tècnics dels que s'haurà de desenvolupar, així com a qualsevol diagrama necessari que aïlli qualsevol dubte que pugui sorgir en el desenvolupament i que pot fer que es replantegin aspectes ja implementats. En aquest cas seran els diagrames de classes i de cassos d'ús i les pantalles de com haurà de ser la interfície gràfica de usuari.

La fase de desenvolupament finalitzarà quan s'obtingui el executable que proporcioni una interfície gràfica que compleixi amb els requisits que es detallen en aquest document.

La fase de documentació finalitzarà amb l'elaboració del present document.

La fase de presentació finalitzarà quan s'elabori i s'exposi una presentació del projecte.

El projecte haurà de seguir les fases en l'ordre exposat i finalitzarà amb l'última fase.

1.3 Resultats esperats

Els resultats esperats i entregats són els següents:

- Aplicació mòbil.
- Memòria del projecte. És el present document que conté tota la informació referent al projecte.
- Vídeo explicatiu.

1.4 Planificació

1.4.1 Planificació inicial

La planificació inicial del projecte es la següent:

| Tarea | Duracio | Inici | Fi |
|----------------|---------|------------|------------|
| Inici Projecte | 0 dies | 16/09/2013 | 16/09/2013 |
| Pac 1 | 7 dies | 23/09/2013 | 01/10/2013 |
| Pac 2 | 28 dies | 02/10/2013 | 29/10/2013 |
| Pac 3 | 43 dies | 30/10/2013 | 10/12/2013 |
| Entrega final | 28 dies | 11/12/2013 | 08/01/2014 |

1.4.2 Planificació real

Es va poder seguir la planificació inicial fins a la PAC2. La PAC3 i la entrega final, per motius de feina, no s'ha pogut seguir la planificació. Per tant, la planificació real queda d'aquesta manera:

| Tarea | Duracio | Inici | Fi |
|----------------|---------|------------|------------|
| Inici Projecte | 0 dies | 16/09/2013 | 16/09/2013 |
| Pac 1 | 7 dies | 23/09/2013 | 01/10/2013 |
| Pac 2 | 28 dies | 02/10/2013 | 29/10/2013 |
| Pac 3 | 55 dies | 30/10/2013 | 25/12/2013 |
| Entrega final | 32 dies | 11/12/2013 | 12/01/2014 |

M'agradaria donar les gràcies al meu tutor per haver-me donat aquesta prorroga de dies.

2. Funcionalitat

2.1 Característiques funcionals.

A continuació es detallaran les funcionalitats finals que tindrà la aplicació:

- L'usuari podrà veure els comentaris que altres usuaris han publicat amb les opinions de les activitats realitzades.
- Podrà veure la fitxa de les activitats que ofereix la cooperativa.
- Podrà cerca una activitat que tingui a prop seu.
- I per finalitzar podrà accedir a consultar el seu espai personal.

Aquesta última funcionalitat no és del tot cert. Ja he comentat que no em vaig poder posar en contacte amb el programador de la cooperativa i per tant, no he pogut consultar cap dada de la seva base de dades, per tant l'espai personal, de moment i mentre no es solucioni aquesta incidència, serà el mateix per tothom.

3. Disseny centrat en l'usuari

Aquesta etapa s'ha realitzat seguint les pautes del disseny centrat a l'usuari.

Aquestes pautes involucren a l'usuari a les fases de desenvolupament del disseny de l'ampliació amb la finalitat de fer el producte final amb les necessitat i característiques que els usuaris necessiten o busquen en aquesta ampliació.

Les fases del CDU són: anàlisi, disseny i avaluació.

A continuació es mostren els punts que treballen aquestes fases.

3.1 Usuaris i context d'ús.

FentPais és una plataforma on els usuaris podran veure les activitats que es fan a prop de la seva ubicació actual, podran veure opinions d'altres usuaris sobre certes activitats i podrà veure la fitxa tècnica de les activitats ofertes per la cooperativa.

Quan es realitza un disseny d'una aplicació és molt important conèixer el perfil del usuaris, quines són les seves necessitats, en quin contextos d'ús utilitzen l'aplicació i els seus objectius. Saber quines funcionalitats desitgen i quines no és primordial per a concretar quines funcionalitats ha d'incorporar el disseny per satisfer les necessitats reals dels usuaris.

S'ha escollit dues tècniques d'indagació per a realitzar la fase de disseny. S'ha exclòs el anàlisi competitiu ja que no hi una aplicació en el market que s'assembli al nostre disseny. La tècnica triada és la següent:

- Observació i investigació contextual (Contextual Enquiry)
Permet conèixer de primera mà com interactuen els usuaris amb el sistema, la qual cosa possibilita la detecció de mancances en el disseny.

3.2 Investigació: Contextual Enquiry

En el market hi ha una aplicació que s'assembla bastant al nostre projecte. L'aplicació es diu tripadvisor. Es demanaran a tres persones que facin tres tasques i que després ens donin la seva opinió.

Les tasques que es demanarà que realitzin els usuaris, són les següents:

- Tasca 1: Registrar-se.
- Tasca 2: Cercar un hotel.
- Tasca 3: Escriure una opinió.

Per a realitzar aquest test s'utilitzarà un dispositiu Samsung S2 amb l'aplicació instal·lada

S'han triat tres persones per a que realitzin el test. Una persona amb un alt nivell, una altra amb nivell mitja i la última amb nivell baix. Les tres persones han realitzat les tasques i les han avaluat i el resultat ha estat el següent:

| Raul Muñoz Vera 28 anys | |
|---------------------------------------|--|
| | |
| Perfil | Estudiant ETIS |
| | |
| Tasca 1 | Sense problemes |
| Tasca 2 | Sense problemes |
| Tasca 3 | Sense problemes |
| Conclusions | Aplicació molt senzilla, però mol útil i molt ben estructurada |

| Josep Vila 28 anys | |
|----------------------------------|----------------------------|
| | |
| Perfil | Mecànic |
| | |
| Tasca 1 | Sense problemes |
| Tasca 2 | Sense problemes |
| Tasca 3 | Sense problemes |
| Conclusions | Li ha agradat l'aplicació. |

| Julian Muñoz | | 58 anys |
|--------------|---|---------|
| Perfil | Jubilat | |
| Tasca 1 | Sense problemes | |
| Tasca 2 | Sense problemes | |
| Tasca 3 | Troba que es tenen que fer masses passos | |
| Conclusions | Li ha agradat que tingues a mà el cercador. | |

3.3 Perfils d'usuari

En una primera instància s'havia pensat en dividir els usuaris de l'aplicació en edat, però com es vol fer una aplicació intuïtiva (que sigui de fàcil ús per a totes les edats), al final s'ha optat en fer aquesta divisió:

- Usuaris actius → Són els usuaris que utilitzen l'aplicació per opinar o per fer reserves d'activitats
- Usuaris passiu → Aquest són els usuaris que només utilitzen l'aplicació per veure les opinions d'altres usuaris. Coneixen la funcionalitat de l'aplicació però no aprofiten tota la seva funcionalitat.

3.4 Context d'ús

El context d'ús es força ampli, ja que es tracta d'una aplicació per a dispositius mòbils.

Com que actualment els smartphones són com petits ordinadors, els usuaris de l'aplicació podran fer servir l'aplicació en qualsevol lloc i en qualsevol moment, sempre i quan disposin d'un dispositiu Android amb connexió a Internet. També serà imprescindible que el dispositiu tingui antena GPS (actualment tots els Smartphones més actuals tenen GPS). Així que no serà cap problema utilitzar l'aplicació.

3.5 Anàlisi de tasques

Les tasques mínimes que implementarà l'aplicació FentPais i que més tard avaluaran els usuaris mitjançant el prototip són les següents:

- Tasca 1. Entrar en el sistema.
 - ⇒ Iniciar l'aplicació.
 - ⇒ Omplir les dades que demanen.
 - ⇒ Prémer botó entrar.

- Tasca 2. Cerca una activitat que estigui a prop del usuari.
 - ⇒ Iniciar l'aplicació.
 - ⇒ Iniciar sessió.
 - ⇒ En la finestra principal clicar sobre a prop meu.

- Tasca 3. Mirar comentaris dels altres usuaris.
 - ⇒ Iniciar l'aplicació.
 - ⇒ Iniciar sessió.
 - ⇒ En la finestra principal clicar sobre Comentaris.

- Tasca 4. Veure la fitxa de una activitat/establiment.
 - ⇒ Iniciar l'aplicació.
 - ⇒ Iniciar sessió.
 - ⇒ En la finestra principal clicar sobre Establiments.
 - ⇒ Clicar sobre la categoria.

- Tasca 5. Entrar al espai personal.
 - ⇒ Iniciar l'aplicació.
 - ⇒ Iniciar sessió.
 - ⇒ En la finestra principal clicar sobre Usuari.

3.6 Escenari d'ús

En el desenvolupament de tasques en els escenaris d'ús es dona per suposat que l'usuari té l'aplicació instal·lada.

Escenari 1

| | |
|----------------------------|-----------------------------------|
| Perfil usuari | Usuari pasiu |
| Context | Qualsevol indret |
| Objectiu | Mirar comentaris d'altres usuaris |
| Tasca | Tasca 1 i tasca 3 |
| Funcionalitats necessàries | Connexió a internet |

Personatges

El Xavier executa l'aplicació FentPais i no està gaire segur de fer una activitat que ofereix la cooperativa; per tant, executa l'aplicació, entra en el sistema i mira els comentaris per veure si hi ha alguna persona que hagi realitzat l'activitat i hagi deixat un comentari.

Escenari 2

| | |
|----------------------------|-----------------------------|
| Perfil usuari | Usuari Actiu |
| Context | Viatge de parella |
| Objectiu | Cercar una activitat |
| Tasca | Tasca 1 i tasca 2 i tasca 4 |
| Funcionalitats necessaries | Connexió a internet |

Personatges

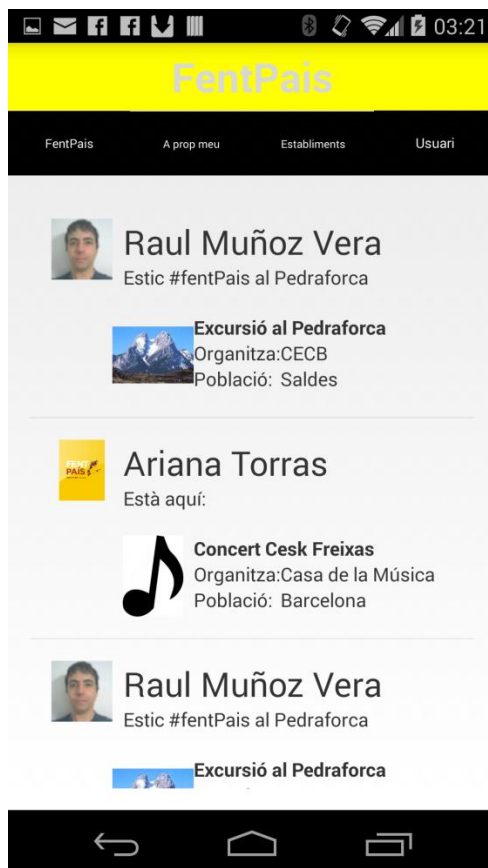
El Josep i la Marta es troben a Manresa, fa un dia molt maco i no saben que fer. Així que obren l'aplicació i busquen una activitat que es pugui realitzar a prop seu. Un cop han trobat una activitat que els pot interessar consulten la fitxa de la activitat.

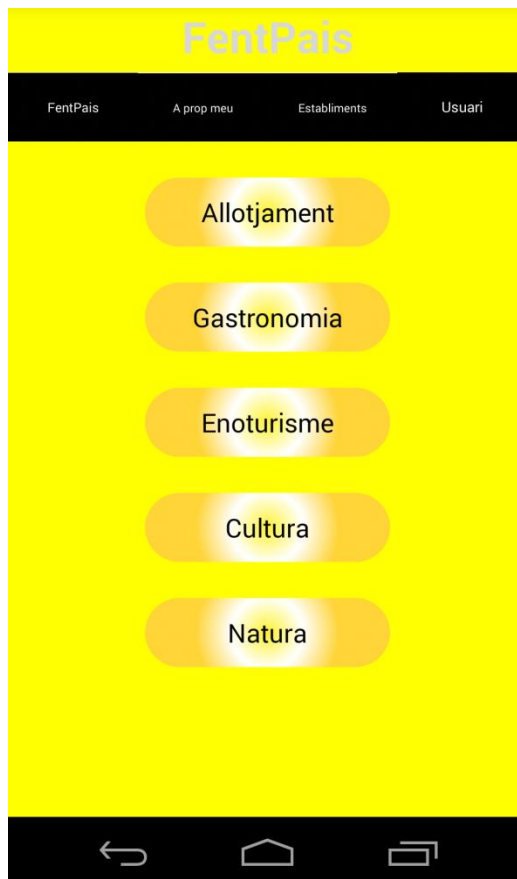
3.7 Prototips horitzontals d'alta fidelitat

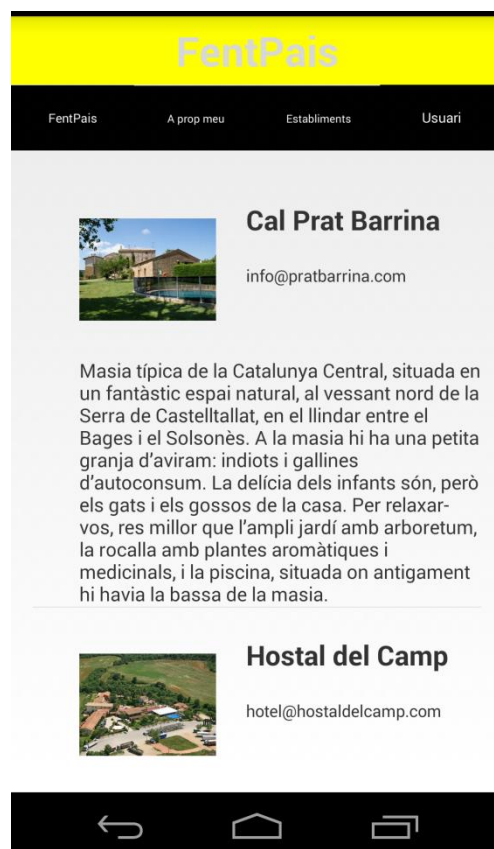
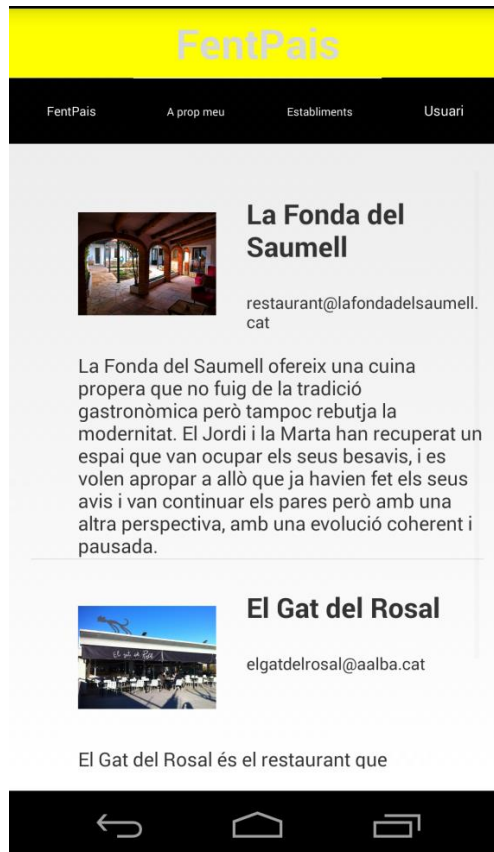
Per a realitzar les pantalles de l'aplicació, primerament es genera sketches (tots els sketches es fan a mà) i després es passa a implementar el prototip.

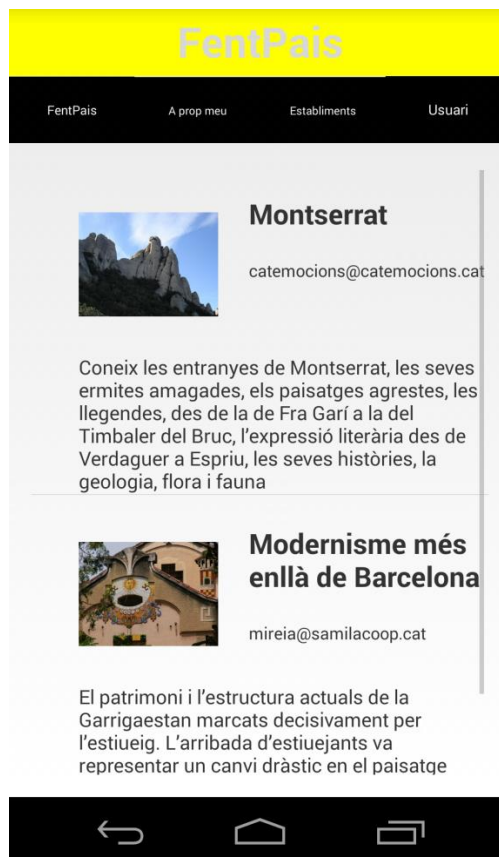
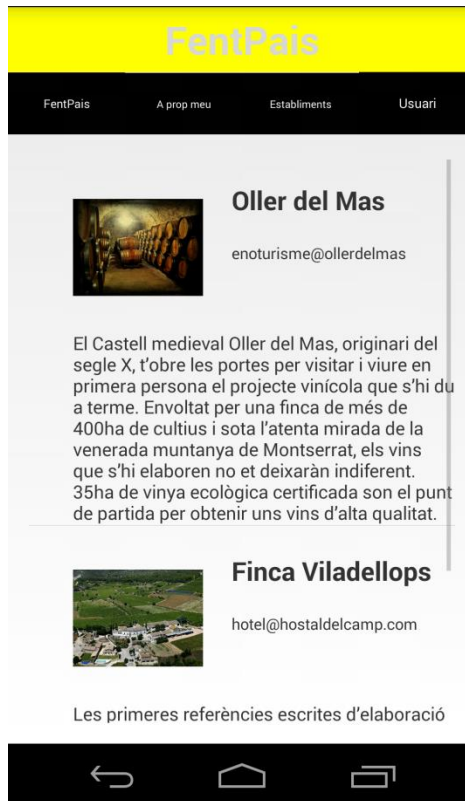
Per a mostrar els prototips, s'ha seguit l'ordre habitual d'utilització:

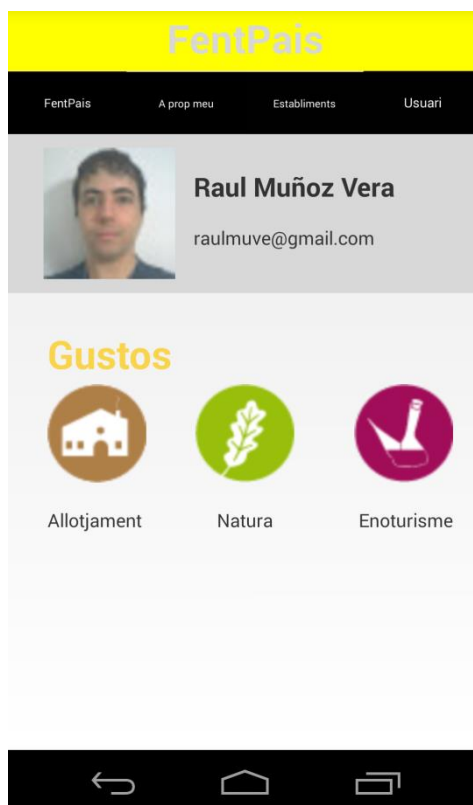
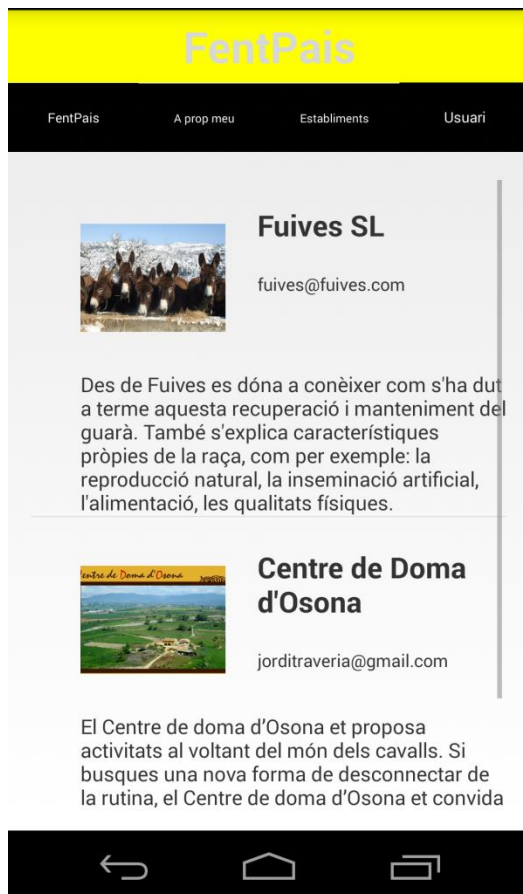












4. Arquitectura de l'aplicació

La funcionalitat principal de l'aplicació és la de realitzar les necessitats que vulgui fer l'usuari. Totes les classes estan implementades dintre d'un paquet anomenat: `com.example.mapafinal`

El paquet té aquest nom, perquè la aplicació fa servir serveis de google, i per tenir accés a aquest serveis tens que especificar el paquet.

Fent unes proves abans de implementar l'aplicació em funcionava amb aquest nom de paquet, per tant no l'he canviat.

L'aplicació conté 15 classes, a continuació es comentaran més a fons cada una d'aquestes classes i es mostrarà una part de la classe.

4.1 Classes del sistema

Seguint un ordre d'ús de l'aplicació, tenim:

4.1.1 Classe PrincipalActivity

Aquesta és la classe amb la qual s'obre l'aplicació, mostra el logotip de la cooperativa durant tres segons i després passa a la classe de validar usuari.

```
public class PrincipalActivity extends Activity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
  
        // La següent línia serveix per eliminar el títol  
        // del projecte  
        requestWindowFeature(Window.FEATURE_NO_TITLE);  
  
        setContentView(R.layout.layout_principal);  
  
        /**  
         * Aquesta pantalla es la principal i es visualitzara durant  
         * 3 segons i despres saltara automaticament a una altra  
         * activitat  
         */  
  
        new Thread() {  
            public void run() {  
                try{  
                    sleep(3000);  
                } catch (Exception e) { }  
  
                Intent i = new Intent(PrincipalActivity.this, Validar.class);  
                startActivity(i);  
  
                finish();  
            }  
        }.start();  
    }  
}
```

4.1.2 Classe Validar

Aquesta classe el que fa es validar el usuari amb una contrasenya.

Actualment només pot validar un usuari, ja que no es connecta a cap base de dades, però en un futur s'implementarà que es connecti a una base de dades externa i comprovi si el usuari ha introduït el usuari i la contrasenya correctes.

```
public class Usuari extends Activity {

    Button btnComentari, btnAProp, btnEstabliments;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        // la següent línia serveix per eliminar el títol
        // del projecte
        requestWindowFeature(Window.FEATURE_NO_TITLE);

        setContentView(R.layout.layout_usuari);

        btnAProp = (Button) findViewById(R.id.btnAprop);
        btnComentari = (Button) findViewById(R.id.btnComentari);
        btnEstabliments = (Button) findViewById(R.id.btnEstabliments);

        btnAProp.setOnClickListener(new OnClickListener() {

            @Override
            public void onClick(View v) {
                Intent i = new Intent(Usuari.this, Mapa.class);
                startActivity(i);
                finish();
            }
        });

        btnComentari.setOnClickListener(new OnClickListener() {

            @Override
            public void onClick(View v) {
                Intent i = new Intent(Usuari.this,
                    LlistaComentaris.class);
                startActivity(i);
                finish();
            }
        });
    }
}
```

4.1.3 Classe LlistaComentaris

Aquesta classe s'encarrega de mostrar per pantalla els comentaris que s'introdueixen a l'aplicació.

El que fa es omplir un ArrayList amb els comentaris que es volen mostrar i llavors crida a un Adapter per mostrar els comentaris per pantalla.

```
public class LlistaComentaris extends Activity {
    private ArrayList<Comentaris> comentaris;
    private ListView lvComentaris;
    private ComentarisAdapter adapter;
    private TextView NomUsuariItem, ComentariItem, TextActivitat,
        EmpresaOrganitza, PoblacioOrganitza;

    //Variables dels botons de la botonera
    Button btnAprop;
    Button btnEstabliments;
    Button btnUsuari;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        requestWindowFeature(Window.FEATURE_NO_TITLE);
        setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);
        setContentView(R.layout.layout_pantalla_comentaris);

        // //Inicialitzem les variables
        comentaris = new ArrayList<Comentaris>();

        OmplirArrayList();

        NomUsuariItem = (TextView) findViewById(R.id.NomUsuariItem);
        ComentariItem = (TextView) findViewById(R.id.ComentarItem);
        TextActivitat = (TextView) findViewById(R.id.TextActivitat);
        EmpresaOrganitza = (TextView) findViewById(R.id.EmpresaOrganitza);
        PoblacioOrganitza = (TextView) findViewById(R.id.PoblacioOrganitza);

        adapter = new ComentarisAdapter(this, comentaris);

        lvComentaris = (ListView) findViewById(R.id.lvItems);

        // Assignem el Adapter al ListView, en aquest punt volem
        // que el ListView mostri les dades que volem.
        lvComentaris.setAdapter(adapter);
    }
}
```

4.1.4 Classe Mapa

Aquesta classe és la que crida als serveis de google; en aquest cas al servei GoogleMaps.

Aquesta classe descarrega el mapa i carrega els marcadors que nosaltres li hem introduït.

```
public class Mapa extends FragmentActivity implements OnMapClickListener {  
  
    GoogleMap mapa;  
    private static final LatLng MANRESA = new LatLng(41.7292826,  
        1.8225153999999293);  
    Button btnComentari;  
    Button btnEstabliments;  
    Button btnUsuari;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        requestWindowFeature(Window.FEATURE_NO_TITLE);  
        setContentView(R.layout.layout_pantalla_principal);  
  
        FragmentManager myFM = getSupportFragmentManager();  
        SupportMapFragment myMAPF = (SupportMapFragment) myFM  
            .findFragmentById(R.id.map);  
  
        mapa = myMAPF.getMap();  
        mapa.setMyLocationEnabled(true);  
        mapa.setMapType(GoogleMap.MAP_TYPE_NORMAL);  
        mapa.moveCamera(CameraUpdateFactory.newLatLngZoom(MANRESA, 8));  
  
        AfegirMarkersAllotjament();  
  
        AfegirMarkersGastronomia();  
  
        AfegirMarkerEnoturisme();  
  
        AfegirMarkerCultura();  
  
        AfegirMarkerNatura();  
  
        btnComentari = (Button) findViewById(R.id.btnComentari);  
  
        btnComentari.setOnClickListener(new OnClickListener() {  
            ..  
            ..  
        });  
    }  
}
```

4.1.5 Classe LlistaEstabliments

Aquesta classe implementa la funcionalitat de mostrar per pantalla els 5 botons de les categories de les activitats.

Cada botó ens enviarà a una llista amb les activitats referents en la categoria seleccionada.

```
public class LlistaEstabliments extends Activity {  
  
    // Botons de la butonera  
    Button btnComentaris, btnAProp, btnUsuari;  
  
    // Botons per triar la opció  
    Button btnAllotj, btnGastro, btnEno, btnNat, btnCultura;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        requestWindowFeature(Window.FEATURE_NO_TITLE);  
        setContentView(R.layout.buttons_establiments);  
  
        btnComentaris = (Button) findViewById(R.id.btnComentari);  
  
        btnComentaris.setOnClickListener(new OnClickListener() {  
  
            @Override  
            public void onClick(View v) {  
                Intent i = new Intent(LlistaEstabliments.this,  
                    LlistaComentaris.class);  
                startActivity(i);  
                finish();  
            }  
        });  
  
        btnAProp = (Button) findViewById(R.id.btnAprop);  
  
        btnAProp.setOnClickListener(new OnClickListener() {  
  
            @Override  
            public void onClick(View v) {  
                Intent i = new Intent(LlistaEstabliments.this, Mapa.class);  
                startActivity(i);  
                finish();  
            }  
        });  
    }  
}
```


4.1.6 Classe ListAllotjaments

Mostra una llista amb els allotjaments que tenim disponibles

```
public class ListAllotjaments extends Activity {

    private ArrayList<Establiment> llistaEstabliments;
    private ListView lvEstabliments;
    private EstablimentsAdapter adapter;
    private TextView nomItemEstabliment, descripcioItemEstabliment,
        comentariItemEstabliment;

    // Variables dels botons de la botonera
    Button btnAprop;
    Button btnComentaris;
    Button btnUsuari;

    @Override
    public void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        requestWindowFeature(Window.FEATURE_NO_TITLE);
        setContentView(R.layout.layout_pantalla_establiments);

        // //Inicialitzem les variables
        llistaEstabliments = new ArrayList<Establiment>();

        OmplirArrayList();

        nomItemEstabliment = (TextView) findViewById(R.id.establimentNom);
        descripcioItemEstabliment = (TextView) findViewById(R.id.establimentDescripcio);
        comentariItemEstabliment = (TextView) findViewById(R.id.TextActivitat);

        adapter = new EstablimentsAdapter(this, llistaEstabliments);

        lvEstabliments = (ListView) findViewById(R.id.lvEstabliments);

        // Assignem el Adapter al ListView, en aquest punt volem
        // que el ListView mostri les dades que volem.
        lvEstabliments.setAdapter(adapter);
    }
}
```

4.1.7 Classe ListCultura

Mostra una llista amb les activitats culturals que estan disponibles.

```
public class ListCultura extends Activity {  
  
    private ArrayList<Establiment> llistaEstabliments;  
    private ListView lvEstabliments;  
    private EstablimentsAdapter adapter;  
    private TextView nomItemEstabliment, descripcioItemEstabliment,  
        comentariItemEstabliment;  
  
    // Variables dels botons de la botonera  
    Button btnAprop;  
    Button btnComentaris;  
    Button btnUsuari;  
  
    @Override  
    public void onCreate(Bundle savedInstanceState) {  
  
        super.onCreate(savedInstanceState);  
        requestWindowFeature(Window.FEATURE_NO_TITLE);  
        setContentView(R.layout.layout_pantalla_establiments);  
  
        // //Inicialitzem les variables  
        llistaEstabliments = new ArrayList<Establiment>();  
  
        OmplirArrayList();  
  
        nomItemEstabliment = (TextView) findViewById(R.id.establimentNom);  
        descripcioItemEstabliment = (TextView) findViewById(R.id.establimentDescripcio);  
        comentariItemEstabliment = (TextView) findViewById(R.id.TextActivitat);  
  
        adapter = new EstablimentsAdapter(this, llistaEstabliments);  
  
        lvEstabliments = (ListView) findViewById(R.id.lvEstabliments);  
    }  
}
```

4.1.8 Classe ListEnoturisme

Mostra una llista amb les activitats sobre enoturisme que podem realitzar.

```
public class ListEnoturisme extends Activity {

    private ArrayList<Establiment> llistaEstabliments;
    private ListView lvEstabliments;
    private EstablimentsAdapter adapter;
    private TextView nomItemEstabliment, descripcioItemEstabliment,
        comentariItemEstabliment;

    // Variables dels botons de la botonera
    Button btnAprop;
    Button btnComentaris;
    Button btnUsuari;

    @Override
    public void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        requestWindowFeature(Window.FEATURE_NO_TITLE);
        setContentView(R.layout.layout_pantalla_establiments);

        // //Inicialitzem les variables
        llistaEstabliments = new ArrayList<Establiment>();

        OmplirArrayList();

        nomItemEstabliment = (TextView) findViewById(R.id.establimentNom);
        descripcioItemEstabliment = (TextView) findViewById(R.id.establimentDescripcio);
        comentariItemEstabliment = (TextView) findViewById(R.id.TextActivitat);

        adapter = new EstablimentsAdapter(this, llistaEstabliments);

        lvEstabliments = (ListView) findViewById(R.id.lvEstabliments);
    }
}
```

4.1.9 Classe ListGastronomia

Mostra una llista dels restaurants que ens recomana la cooperativa FentPais.

```
public class ListGastronomia extends Activity {

    private ArrayList<Establiment> llistaEstabliments;
    private ListView lvEstabliments;
    private EstablimentsAdapter adapter;
    private TextView nomItemEstabliment, descripcioItemEstabliment,
        comentariItemEstabliment;

    // Variables dels botons de la botonera
    Button btnAprop;
    Button btnComentaris;
    Button btnUsuari;

    @Override
    public void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        requestWindowFeature(Window.FEATURE_NO_TITLE);
        setContentView(R.layout.layout_pantalla_establiments);

        // //Inicialitzem les variables
        llistaEstabliments = new ArrayList<Establiment>();

        OmplirArrayList();

        nomItemEstabliment = (TextView) findViewById(R.id.establimentNom);
        descripcioItemEstabliment = (TextView) findViewById(R.id.establimentDescripcio);
        comentariItemEstabliment = (TextView) findViewById(R.id.TextActivitat);

        adapter = new EstablimentsAdapter(this, llistaEstabliments);

        lvEstabliments = (ListView) findViewById(R.id.lvEstabliments);
```

4.1.10 Classe ListNatura

Mostra una llista amb les activitats naturals que podem realitzar.

```
public class ListNatura extends Activity {

    private ArrayList<Establiment> llistaEstabliments;
    private ListView lvEstabliments;
    private EstablimentsAdapter adapter;
    private TextView nomItemEstabliment, descripcioItemEstabliment,
        comentariItemEstabliment;

    // Variables dels botons de la botonera
    Button btnAprop;
    Button btnComentaris;
    Button btnUsuari;

    @Override
    public void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        requestWindowFeature(Window.FEATURE_NO_TITLE);
        setContentView(R.layout.layout_pantalla_establiments);

        // //Inicialitzem les variables
        llistaEstabliments = new ArrayList<Establiment>();

        OmplirArrayList();

        nomItemEstabliment = (TextView) findViewById(R.id.establimentNom);
        descripcioItemEstabliment = (TextView) findViewById(R.id.establimentDescripcio);
        comentariItemEstabliment = (TextView) findViewById(R.id.TextActivitat);

        adapter = new EstablimentsAdapter(this, llistaEstabliments);

        lvEstabliments = (ListView) findViewById(R.id.lvEstabliments);
```

4.1.11 Classe Usuari

Aquesta classe es la encarregada de mostrar el perfil d'usuari.

Com ja em comentat, actualment només mostra un usuari, però en un futur s'implementarà que es descarregui les dades d'un servidor extern i les mostri per pantalla.

```
public class Usuari extends Activity {

    Button btnComentari, btnAProp, btnEstabliments;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        // La següent línia serveix per eliminar el títol
        // del projecte
        requestWindowFeature(Window.FEATURE_NO_TITLE);

        setContentView(R.layout.layout_usuari);

        btnAProp = (Button) findViewById(R.id.btnAprop);
        btnComentari = (Button) findViewById(R.id.btnComentari);
        btnEstabliments = (Button) findViewById(R.id.btnEstabliments);

        btnAProp.setOnClickListener(new OnClickListener() {

            @Override
            public void onClick(View v) {
                Intent i = new Intent(Usuari.this, Mapa.class);
                startActivity(i);
                finish();
            }
        });

        btnComentari.setOnClickListener(new OnClickListener() {

            @Override
            public void onClick(View v) {
                Intent i = new Intent(Usuari.this,
                    LlistaComentaris.class);
                startActivity(i);
                finish();
            }
        });
    }
}
```

4.1.12 Classe ComentarAdapter

Aquesta classe es la encarregada de adaptar cada comentari a la llista.

```
public class ComentarAdapter extends ArrayAdapter<Comentaris> {
    private Context context;
    private ArrayList<Comentaris> datos;

    public ComentarAdapter(Context context, ArrayList<Comentaris> datos) {
        super(context, R.layout.comentaris_item, datos);
        // Guardamos los parámetros en variables de clase.
        this.context = context;
        this.datos = datos;
    }

    @Override
    public View getView(int position, View convertView, ViewGroup parent) {

        View item = LayoutInflater.from(context).inflate(
            R.layout.comentaris_item, null);

        // Assignem el Nom al Item
        TextView nom = (TextView) item.findViewById(R.id.NomUsuariItem);
        nom.setText(datos.get(position).getNom());

        // Assignem la ubicacio
        TextView ubicacio = (TextView) item.findViewById(R.id.ComentarItem);
        ubicacio.setText(datos.get(position).getUbicacio());

        // Assignem Activitat
        TextView activitat = (TextView) item.findViewById(R.id.TextActivitat);
        activitat.setText(datos.get(position).getActivitat());

        // Assignem Empresa
        TextView empresa = (TextView) item.findViewById(R.id.EmpresaOrganitza);
        empresa.setText(datos.get(position).getOrganitza());

        // Assignem Poblacio
        TextView poblacio = (TextView) item
            .findViewById(R.id.PoblacioOrganitza);
        poblacio.setText(datos.get(position).getPoblacio());
    }
}
```

4.1.13 Classe Comentaris

Classe auxiliar que s'utilitza per a crear comentaris.

```
public class Comentaris {  
  
    private String nom, ubicacio, activitat, organitza, poblacio;  
    private int drawableImageID, drawableImageActivity;  
  
    public Comentaris(String nom, String ubicacio, String activitat,  
        String organitza, String poblacio, int drawableImageActivity, int id) {  
  
        this.nom = nom;  
        this.ubicacio = ubicacio;  
        this.activitat = activitat;  
        this.organitza = organitza;  
        this.poblacio = poblacio;  
        this.drawableImageActivity = drawableImageActivity;  
        this.drawableImageID = id;  
    }  
  
    public String getNom() {  
  
        return nom;  
    }  
  
    public String getUbicacio() {  
        return ubicacio;  
    }  
  
    public void setUbicacio(String ubicacio) {  
        this.ubicacio = ubicacio;  
    }  
  
    public String getActivitat() {  
        return activitat;  
    }  
  
    public void setActivitat(String activitat) {  
        this.activitat = activitat;  
    }  
}
```


4.1.14 Classe EstablimentsAdapter

Aquesta classe es la encarregada de adaptar cada establiment a la llista.

```
public class EstablimentsAdapter extends ArrayAdapter<Establiment> {
    private Context context;
    private ArrayList<Establiment> datos;

    public EstablimentsAdapter(Context context, ArrayList<Establiment> datos) {
        super(context, R.layout.comentaris_item, datos);
        // Guardamos los parámetros en variables de clase.
        this.context = context;
        this.datos = datos;
    }

    @Override
    public View getView(int position, View convertView, ViewGroup parent) {

        View item = LayoutInflater.from(context).inflate(
            R.layout.establiment_item, null);

        TextView nom = (TextView) item.findViewById(R.id.establimentNom);
        nom.setText(datos.get(position).getNom());

        TextView descripcio = (TextView) item.findViewById(R.id.establimentInfo);
        descripcio.setText(datos.get(position).getComentari());

        TextView info = (TextView) item.findViewById(R.id.establimentDescripcio);
        info.setText(datos.get(position).getInfo());

        ImageView imgUsuari = (ImageView) item.findViewById(R.id.imgStabliment);
        imgUsuari.setImageResource(datos.get(position).getDrawableImageID());

        return item;
    }
}
```

4.1.15 Classe Establiment

Classe per a crear establiments i després guardar-los al ArrayList.

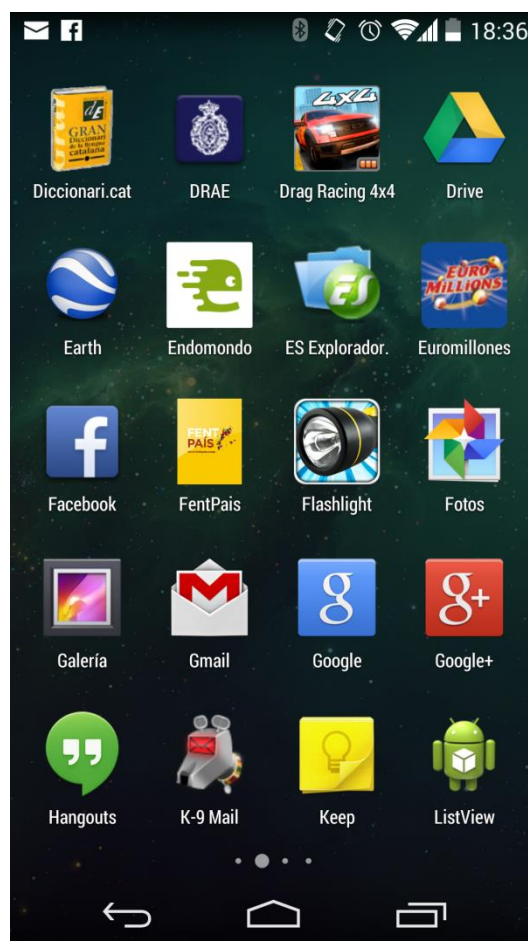
```
public class Establiment {  
  
    private String nom, info, comentari;  
    private int drawableImageID;  
  
    public Establiment(String nom, String info, String comentari,  
        int id) {  
  
        this.nom = nom;  
        this.info = info;  
        this.comentari = comentari;  
        this.drawableImageID = id;  
    }  
  
    public String getNom() {  
        return nom;  
    }  
  
    public void setNom(String nom) {  
        this.nom = nom;  
    }  
  
    public String getInfo() {  
        return info;  
    }  
  
    public void setUbicacio(String ubicacio) {  
        this.info = ubicacio;  
    }  
  
    public String getComentari() {  
        return comentari;  
    }  
  
    public void setComentari(String comentari) {  
        this.comentari = comentari;  
    }  
}
```

5. Disseny interfície gràfica

Per al disseny final de l'aplicació, em tractat de seguir el que es va proposar als prototips inicials, però s'han canviat algunes coses com per exemple, s'han ficat els botons a la part superior ja que en un principi es van col·locar a la part inferior però per recomanació del consultor al final s'han posat a la part superior.

5.1 FentPais

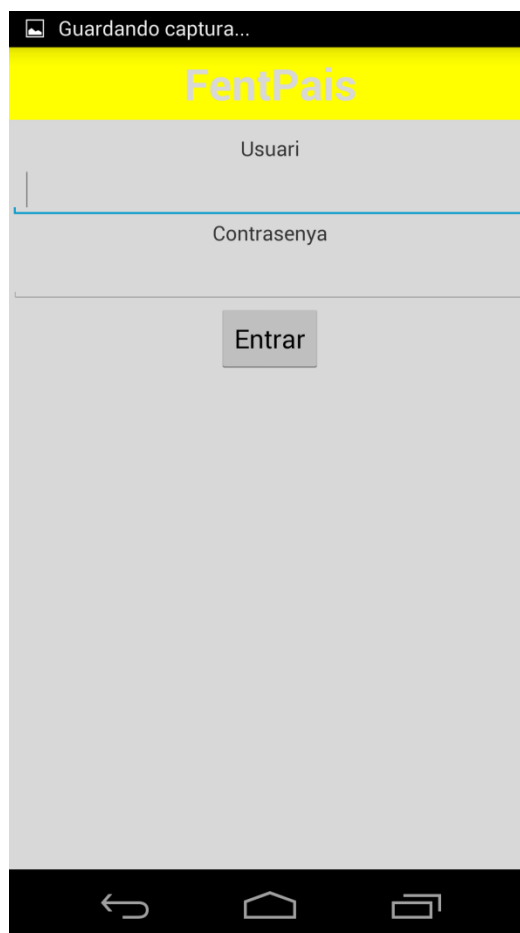
Un cop es selecciona l'aplicació, s'obre una pantalla durant 3 segons del logotip de l'aplicació. Aquesta pantalla, internament no fa res, només es per mostrar el logotip de la cooperativa.





5.2 Validar Usuari

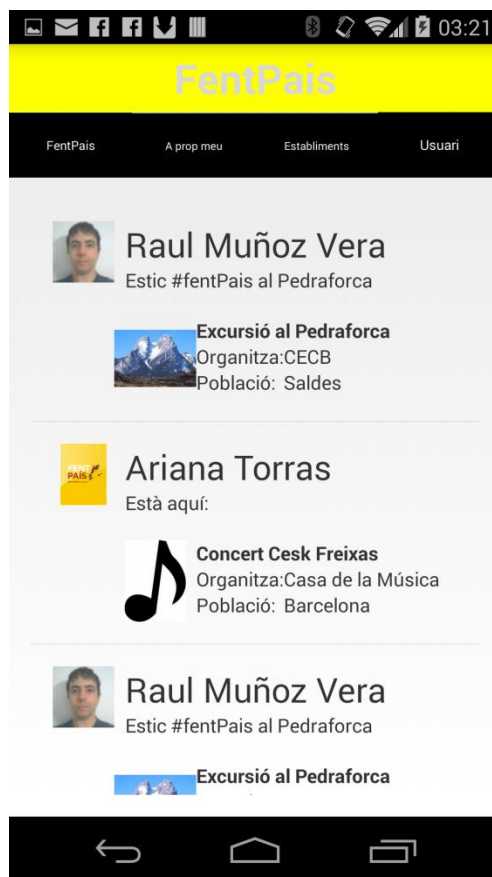
La següent pantalla que ens mostra l'aplicació és la de validar un usuari. Aquesta pantalla verifica que haguem ficat bé el nom d'usuari i la contrasenya. Ara només accepta un usuari i una contrasenya (uoc/uoc) però en un futur es vol que es connecti a una base de dades externa. Si introduïm els camps malament, mostra un missatge d'error i posa els camps en blanc.



5.3 Pantalla comentaris

Un cop ens em validat, se'ns obre la pantalla dels comentaris que ha fet altres usuaris de activitats que han fet.

A la part superior tenim els botons dels apartats de la aplicació (Comentaris, A prop meu, Establiments, Usuari)



5.4 Pantalla A prop meu

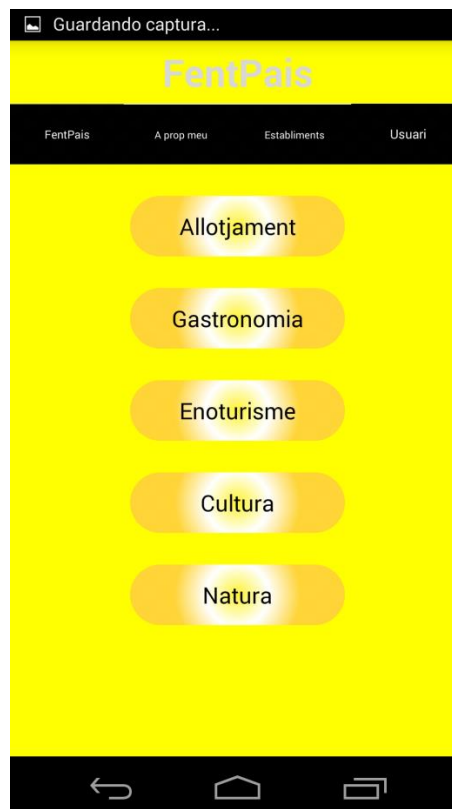
Si accedim a aquesta pantalla l'aplicació ens mostrarà les activitats que tenim mes a prop de la nostra ubicació.

Aquesta pantalla mostra les diferents activitats amb diferents marcadors segons la seva categoria.



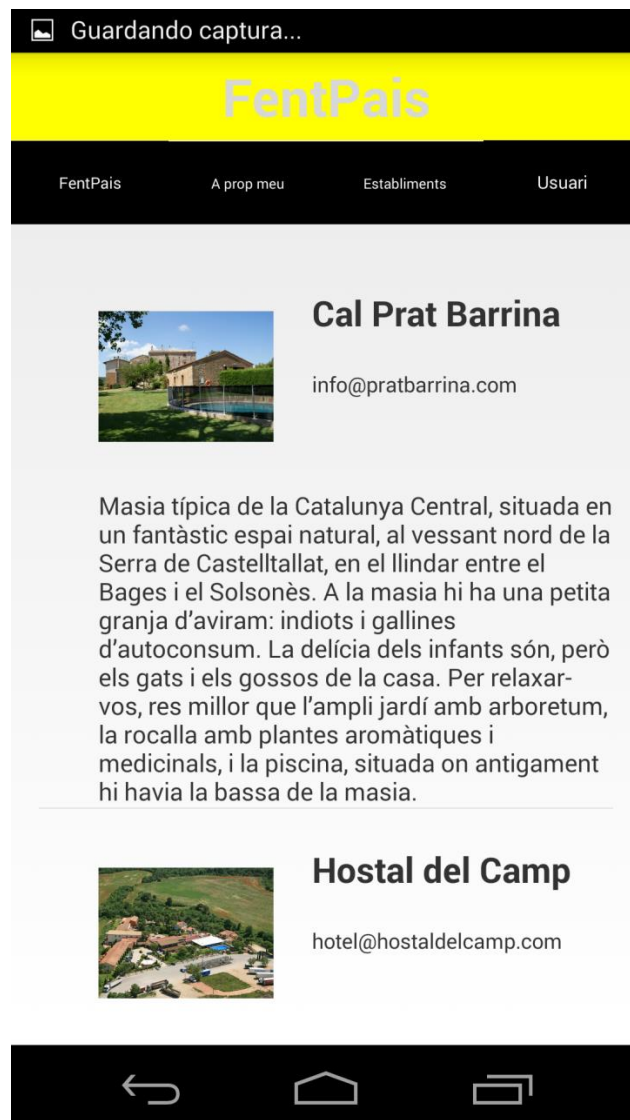
5.5 Pantalla establiments

La següent funcionalitat que trobem a l'aplicació és la pantalla establiments. Aquesta pantalla ens mostra les 5 categories de les activitats per a que podem triar una.



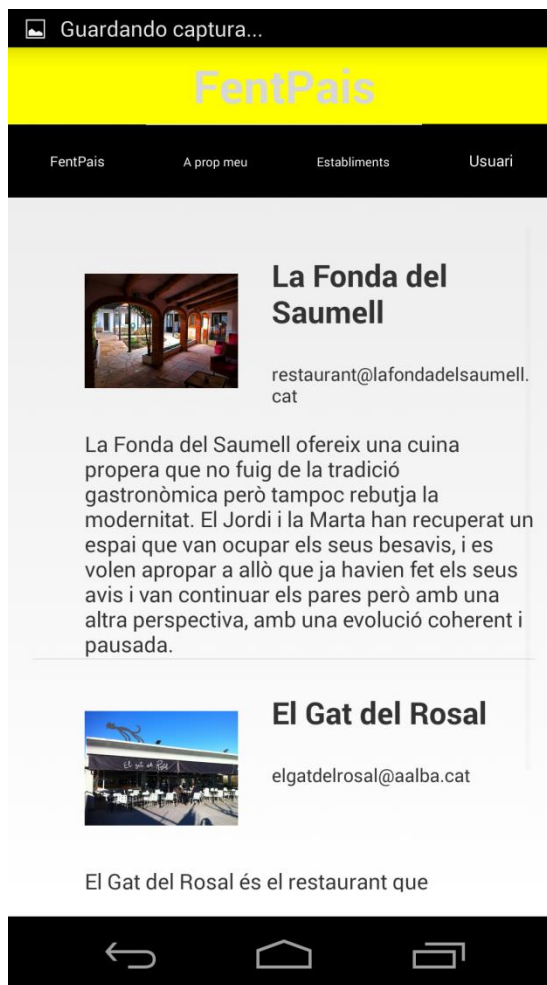
5.6 Pantalla Llista Allotjaments

Aquesta pantalla ens mostra les activitats ofertes referents a Allotjament.



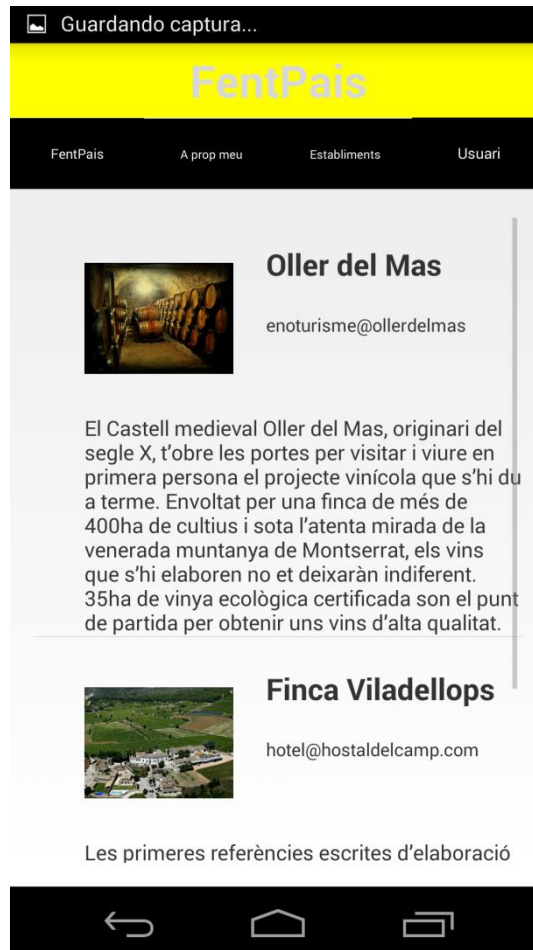
5.7 Pantalla Llista Gastronomia

Aquesta pantalla ens mostra les activitats ofertes referents a gastronomia.



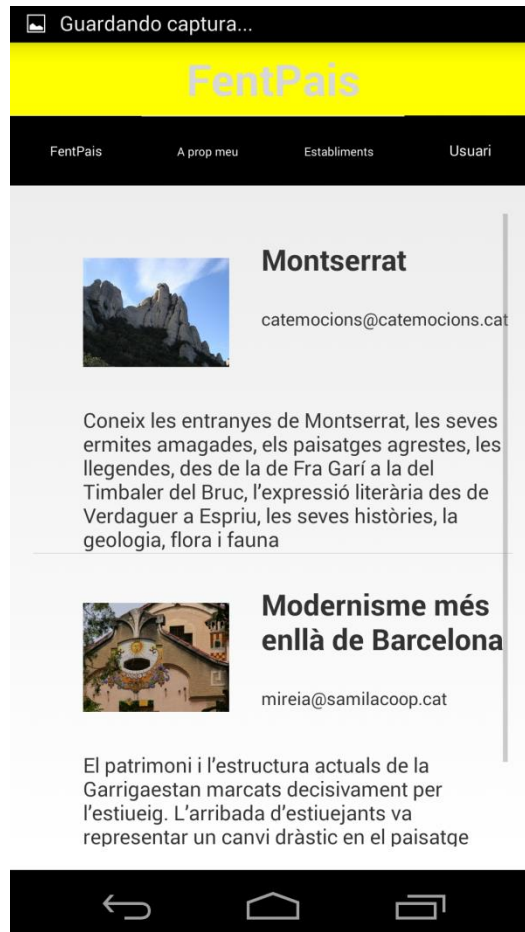
5.8 Pantalla Llista Enoturisme

Aquesta pantalla ens mostra les activitats ofertes referents a Enoturisme.



5.9 Pantalla Llista Cultura

Aquesta pantalla ens mostra les activitats ofertes referents a cultura.



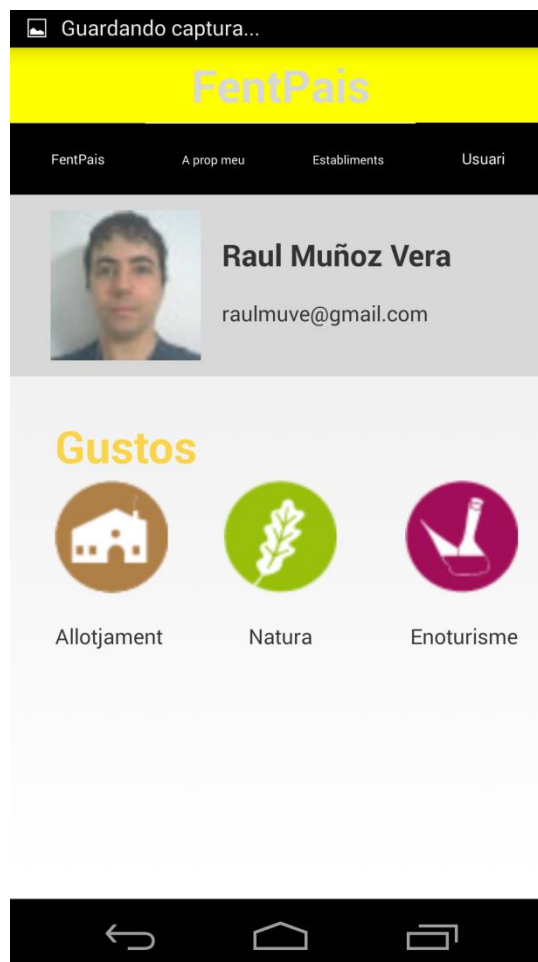
5.10 Pantalla Llista Natura

Aquesta pantalla ens mostra les activitats ofertes referents a activitats relacionades amb la naturalesa.



5.11 Pantalla usuari

Aquesta pantalla mostra els gustos de l'usuari , la seva fotografia i les seves dades personals.



6. Avaluació de la aplicació

Aquesta aplicació va començar amb molta força, moltes idees i sobre tot amb molt temps lliure.

M'hagués agradat una mica més de col·laboració/interès per part de la cooperativa, però per algun motiu o per un altre no ha estat així. No hi hagut molta comunicació i no he pogut fer tot el que a mi m'hagués agradat. També es veritat que la feina m'ha tret molt de temps i no he pogut dedicar tot el temps que a mi m'hagués agradat per fer l'aplicació.

La aplicació funciona correctament i fa tot el que li demanes (dintre de les limitacions que implica no esta connectada a una base de dades), per això estic bastant satisfet del resultat, ja que es la meva primera aplicació android, però tot i així encara es té que millorar molt.

7. Variacions respecte a la proposta inicial

S'han realitzat forces canvis respecte a la proposta inicial. Alguns per recomanació del consultor com pot ser posar la barra dels botons a la part superior, com uns altres que no s'han pogut implementar per falta de temps i/o coneixements; com poden ser per exemple convidar a altres usuaris a través de Facebook o implementar l'agenda.

Però d'altra banda s'ha intentat seguir les pautes que es van proposar des un bon principi.

8. Líneas futuras de treball

Aquesta aplicació encara es té que millorar molt i el primer pas que és té que fer si es vol penjar al market es connectar-la amb la base de dades per a que agafi els comentaris, els usuaris i totes les activitats que ofereix la cooperativa

També s'hauria de implementar la possibilitat de que un usuari pogués convidar a altres usuaris i que pogués comentar les activitats realitzades.

En un futur també m'agradaria implementar el tema de l'agenda, ja que aquest com ho he estat mirant i he fet algunes proves i no m'he arribat a sortir, per això és un tema que tinc pendent i que m'ho miraré.

9. Bibliografía

Web:

<http://android.developers.com/>

<http://www.elandroidelibre.com/2013/10/desarrollando-en-android-2-google-maps-api.html>

<http://www.androidcurso.com/index.php/tutoriales-android/41-unidad-7-seguridad-y-posicionamiento/223-google-maps-api-v2>

<http://proyectosimio.wordpress.com/2013/04/27/programacion-android-recursos-xml-ii/>

<http://proyectosimio.wordpress.com/2013/04/27/programacion-android-recursos-xml-iii/>

<http://www.androidcurso.com/index.php/tutoriales-android/34-unidad-3-actividades-e-intenciones/128-la-vista-listview>

<http://www.programacion.com/articulo/introduccion-a-los-layouts-para-android-400>

<http://www.inforjmr.es/?p=531>

Youtube:

<http://www.youtube.com/watch?v=2zJD4ASpFW8&list=PL7EA29F3B739286CA>

<http://www.youtube.com/watch?v=i1ha4ULoVOw&list=PL7EA29F3B739286CA>

<http://www.youtube.com/watch?v=WleEwzQsi1k&list=PL7EA29F3B739286CA>

<http://www.youtube.com/watch?v=5sdgcHQMrnk&list=PL7EA29F3B739286CA>

<http://www.youtube.com/watch?v=cD1gMRjK1U8&list=PL7EA29F3B739286CA>

<http://www.youtube.com/watch?v=nN9MvWQSZUo>

http://www.youtube.com/watch?v=_Bysl7V99Yk

<http://www.youtube.com/watch?v=a0Qao8Cu5Go>

<http://www.youtube.com/watch?v=7aPGuqJim6c>

<http://www.youtube.com/watch?v=kZYWqL0kaAM&list=PL438E15276487C03D>

<http://www.youtube.com/watch?v=VlSxyEhjGEE&list=SP438E15276487C03D>

<http://www.youtube.com/watch?v=iR6VGBKteu0&list=SP438E15276487C03D>

<http://www.youtube.com/watch?v=OqDArYMWoYc&list=SP438E15276487C03D>

