

# petraCAT



Treball de final de carrera  
*XML i web semàntica*

**Professor:** *Joan Casas Roma*  
**Alumne:** *Oscar Sòria Comas*

Enginyeria tècnica en informàtica de sistemes  
Universitat Oberta de Catalunya

# Índex de Contingut

Presentació del problema .....	1
Objectius del projecte .....	2
La web semàntica .....	3
Evolució de la web.....	3
Inicis de la web.....	3
Web 2.0.....	3
Web semàntica.....	4
Les ontologies.....	4
Antecedents.....	6
Les tecnologies .....	7
XML.....	7
XML Schema.....	7
RDF i OWL.....	8
Exemple de definició de classes i atributs.....	8
Exemple de creació d'instàncies.....	8
XQuery.....	9
Java-Android.....	10
Una visió a gran escala.....	11
L'ontologia .....	12
Jerarquia de classes.....	12
Relacions entre classes.....	13
Atributs de les classes.....	14
Els documents .....	15
Llista de vies .....	15
XML Schema.....	15
Exemple de document.....	16
Informació d'una via.....	17
XML Schema.....	17
Exemple de document.....	20
Les consultes xQuery.....	22
Exemple de consulta.....	22
Crida a una consulta xQuery.....	23
L'aplicació .....	24
Marc d'execució.....	24
Una vista general.....	24
Diagrama de classes.....	29
Diagrama de seqüències (seguiment d'una cerca).....	30
Línies de continuació del projecte .....	32
Conclusions .....	34
Glossari .....	35
Bibliografia.....	37
Annex: Instal·lació del projecte.....	38
Instal·lació de la JVM.....	38

Instal·lació per GNU/Linux.....	38
Instal·lació per Windows o altres.....	38
Instal·lació del gestor de base de dades eXist.....	38
Instal·lació per Windows Vista.....	39
Instal·lació per GNU/Linux o altes versions de Windows.....	39
Instal·lació de l'eclipse.....	39
Instal·lació per GNU/Linux.....	39
Instal·lació per Qualsevol plataforma.....	39
Instal·lació de l'SDK d'Android.....	40
Instal·lació de l'ADT Plugin for Eclipse.....	40
Instal·lació.....	40
Configuració.....	41
Desplegament del projecte.....	43
Ontologia.....	43
Consultes XQuery.....	44
Eclipse.....	44
Execució del programa.....	45
Enviar coordenades GPS a l'emulador.....	45
Pòster del diagrama de classes .....	47

## Presentació del problema

En el món de l'escalada contínuament s'estan obrint vies noves. En alguns casos no es ressenyen i en altres és el propi oberturista qui fa una petita guia que només es pot aconseguir al refugi, càmping o “bar d'escaladors” de la zona. En el cas de les vies més conegudes o les que ja fa un cert temps que s'escalen i que no s'han abandonat, acostumen a sortir guies, de venda a llibreries, que les ressenyen, però malauradament acaben quedant obsoletes per la poca freqüència d'actualització.

Qui vol tenir informació de darrerà mà (noves vies, estat de les que porten més temps obertes, etc.) només té l'opció d'accedir a Internet i fer servir els cercadors normals per buscar el blog d'algun escalador que hagi fet aquella via últimament o adreçar-se a algun fòrum per demanar informació a la resta de participants.

Tot i que s'ha pintat tot molt negre, també s'ha de dir que hi han alguns blogs com <http://escalatroncs.wordpress.com/> que intenten fer un recull de vies i faciliten la seva cerca organitzant-les per diferents tipus de categoria (zona, dificultat...). També existeix un cercador <http://www.ressenya.net/> que s'encarrega de mantenir una base de dades de vies d'escalada a Catalunya. Per desgracia tampoc s'actualitza gaire sovint (avui fa gairebé 5 mesos que van escriure l'última notícia).

Amb la bona intenció d'organitzar i facilitar la cerca d'informació en aquesta petita àrea de coneixement, neix el projecte *petraCAT* que intentarà, per un costat, estudiar l'ús de les ontologies per modelar el coneixement existent sobre les vies d'escalada en roca i, per un altre costat, aprofitar la potencia dels nous dispositius mòbils per poder accedir a la informació d'aquestes vies de la forma més fàcil possible i en mobilitat.

## Objectius del projecte

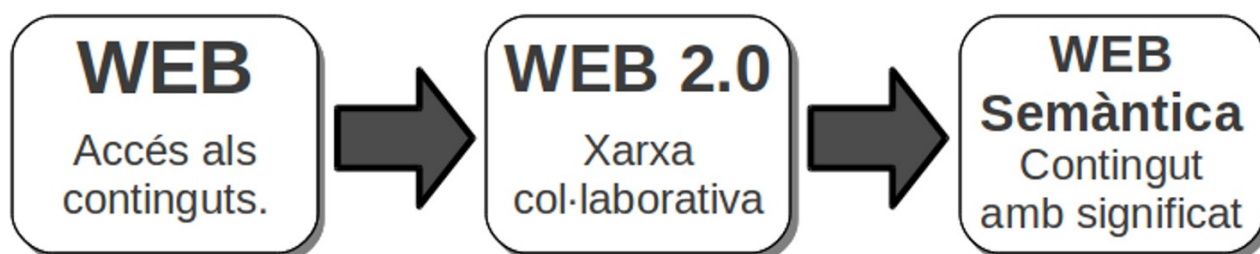
- Conèixer què es la web semàntica.
- Estudiar les tecnologies XML.
- Aprendre el necessari per poder dissenyar una petita ontologia.
- Familiaritzar-se amb les bases de dades XML, un tipus de base de dades diferent de les bases de dades relacionals, les que s'estudien durant l'Enginyeria tècnica.
- Adquirir els coneixements necessaris per poder fer aplicacions per dispositius mòbils.
- Utilitzar tots aquest nous coneixements per donar solució al problema presentat en l'apartat anterior.

## La web semàntica

L'arribada de la web semàntica suposarà un gran salt qualitatiu en el tractament dels continguts a tota la xarxa. Aquesta és una afirmació fàcil de fer si es té en compte com a evolucionat la web des dels seus inicis.

### Evolució de la web

La web, des del seu naixement, ha evolucionat considerablement. Va haver un punt d'inflexió en el pas del que era la web original a la *web 2.0*, i ara estem a les portes d'una nova revolució, el salt a la *web semàntica*.



### Inicis de la web

Es podria dir que la web va néixer al 1989 amb la creació, per part de Tim Berners-Lee, del llenguatge HTML (HyperText Markup Language). Poc després, amb l'ajud de Robert Cailliau, un company de feina del CERN, va desenvolupar el primer navegador, que anomenaria *WorldWideWeb*, i el primer servidor, el *httpd* (hypertext transfer protocol daemon). Per fi, a l'agost del 1991 el CERN va connectar a la xarxa el primer servidor http.

En aquells inicis la web només era contingut estàtic, que podia ser accessible per tothom que tingués una connexió a Internet i un navegador web, i que només podia ser generat per uns pocs autors.

### Web 2.0

La terminologia *Web 2.0* es va començar a fer servir al 2004 com a conseqüència d'una conferència de Tim O'Reilly (president de l'editorial O'Reilly i impulsor del software lliure) en la que parlava d'aquest tema.

A la *web 2.0* no només hi han pàgines estàtiques, també hi han continguts multimèdia, i el més significatiu, han aparegut les xarxes socials. Ara tothom és consumidor i generador de continguts, la xarxa està viva.

Però aquesta nova forma de viure la xarxa a portat també alguns problemes. Ara hi han molts

continguts, tants que poden arribar a saturar a qualsevol, a més, no tot el contingut es de qualitat i els usuaris han d'aprendre a triar què pot ser bo i què no.

## Web semàntica

La nova revolució de la web, encapçalada un altre cop per Tim Berners-Lee des del W3C<sup>1</sup> (World Wide Web Consortium) intenta donar significat al contingut perquè no només els humans siguem capaços d'interpretar-lo, sinó que també ho siguin les màquines.

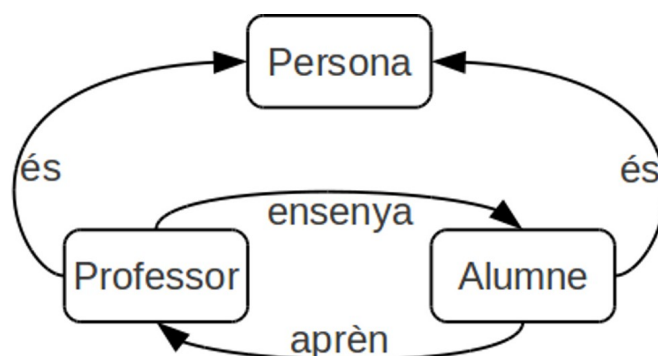
Un dels inconvenients que té la web fins ara, i que intenta solucionar la *web semàntica*, és que en el moment de fer una cerca a Internet, el cercador no pot entendre el que li estem demanant, simplement es limita a cercar l'aparició de les paraules que li hem donat. Això vol dir que si li demanem on podem comprar un televisor nou, ens mostrarà aquesta informació, però també és possible que surti informació sobre programes de televisió o botigues on podem fer compres que no tenen res a veure amb els televisors.

Amb la *web semàntica* els resultats de les cerques fetes haurien de ser molt més precises i no mostrar continguts que no s'han demanat només perquè contenen alguna de les paraules que hem fet servir a la cerca.

## Les ontologies

Les ontologies aportaran la semàntica tan desitjada a la nova web, encara que no és la seva única utilitat, també tenen aplicació en el món de la intel·ligència artificial, en la biomedicina...

Una ontologia té com a objectiu descriure una àrea del coneixement humà. Cada concepte està representat per una classe, aquestes classes es poden relacionar entre elles de diverses maneres i algunes poden derivar d'altres. Per exemple:



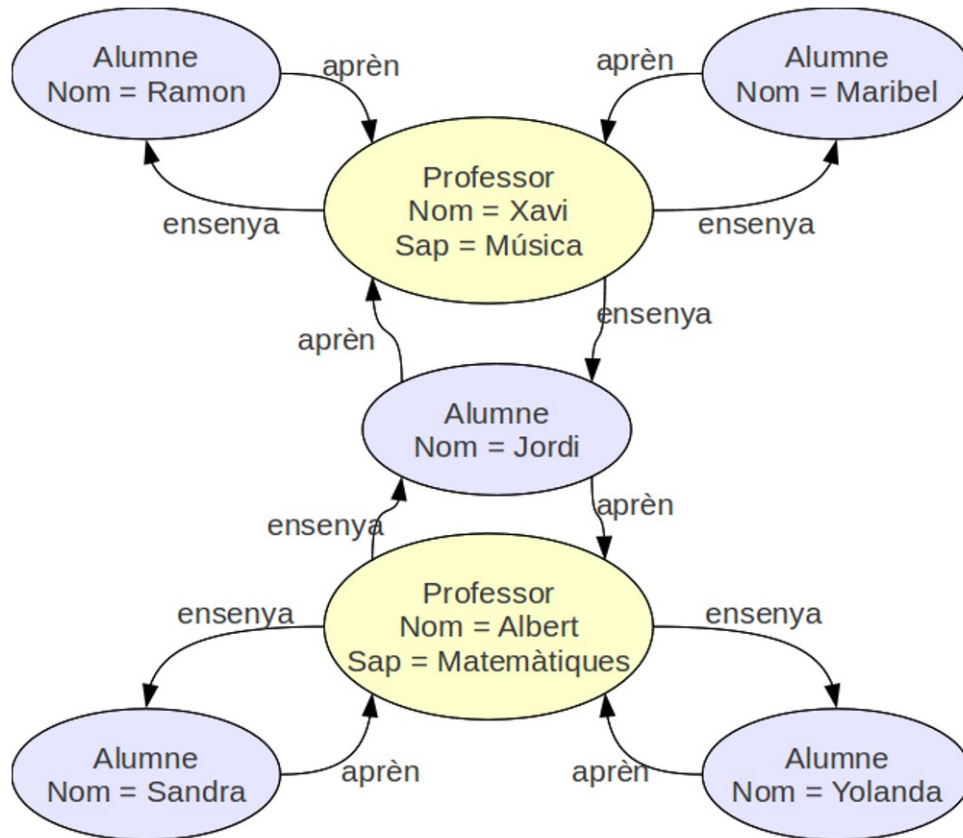
Exemple d'ontologia: Classes

A la imatge es pot veure que les classes Professor i Alumne deriven de la classe Persona i que, a la vegada, estan relacionades entre elles: un professor pot ensenyar a molts alumnes i un alumne pot aprendre de molts professors.

Però una classe és un concepte molt general. Alumnes i professors hi han molts i no tots els professors ensenyen a tots els alumnes, ni tots els alumnes aprenen de tots els professors, així que

<sup>1</sup> Organisme encarregat de definir estàndards per a la web

serà necessari crear instàncies (elements concrets de cada classe) i veure com es relaciona cada professor individual amb cada alumne individual. Per poder concretar aquestes instàncies es faran servir els atributs, tal com mostra la imatge següent:



*Exemple d'ontologia: Instàncies*

A l'exemple es pot veure que: el professor Xavi sap música i ensenya als alumnes Ramon, Maribel i Jordi; el professor Albert sap matemàtiques i ensenya als alumnes Jordi, Sandra i Yolanda; l'alumne Jordi aprèn tan del professor Xavi com del professor Albert...

Per poder representar aquestes ontologies en un ordinador i que els programes les puguin entendre, s'han pensat llenguatges com l'*RDF* i l'*OWL* i eines que faciliten el seu disseny per part del humans com *Protégé*, *Kaon*, *WebOnto*...



## Antecedents

No s'ha trobat cap precedent d'ontologia dins del món de l'escalada.

En el cas del programa cercador sí que hi ha alguna cosa feta: <http://www.ressenya.net> d'on s'han agafat algunes idees per aquest projecte.

## Les tecnologies

### XML

---

XML (eXtensible Markup Language) és un llenguatge d'etiquetes genèric per poder crear qualsevol tipus de document definint la seva sintaxi. Es podria escriure en XML un dels professors de l'exemple anterior, així:

```
<professor nom="Xavi">
  <sap>Música</sap>
</professor>
```

On *<professor>* és l'etiqueta principal del document, *nom* és un atribut i *<sap>* és una etiqueta sotmesa jeràrquicament a *<professor>*. D'això es pot deduir que XML estructura la informació de manera jeràrquica.

Una altra característica que té aquest llenguatge és la seva facilitat per estendre un document. Si es volgués dir que el professor Xavi es especialista en piano, només caldria afegir una etiqueta nova:

```
<professor nom="Xavi">
  <sap>Música</sap>
  <especialitat>piano</especialitat>
</professor>
```

### XML Schema

---

XML Schema és un llenguatge escrit en XML i utilitzat per definir i restringir l'estructura d'un document XML. Es podria definir un document per descriure un professor d'aquesta manera:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- tipus professor -->
  <xs:complexType name="professortype">
    <xs:sequence>
      <xs:element name="sap" type="xs:string" />
      <xs:element name="especialitat" type="xs:string" />
    </xs:sequence>
    <xs:attribute name="nom" type="xs:string" use="required" />
  </xs:complexType>
  <!-- element principal -->
  <xs:element name="professor" type="professortype" />
</xs:schema>
```

## RDF i OWL

RDF (Resource Description Framework) és un llenguatge escrit sobre XML i que aporta el significat semàntic del que manca aquest últim. Per fer-ho, utilitza el triplet **subjecte-predicat-objecte**, on el subjecte és l'element que volem descriure (un professor), el predicat és la propietat que estem definint sobre el subjecte (ensenya) i l'objecte és el valor del predicat per aquell subjecte (un alumne).

OWL (Ontology Web Language) és un llenguatge basat en RDF, al qual afegeix la possibilitat de definir classes, propietats i com s'han de poder relacionar aquestes classes.

### Exemple de definició de classes i atributs

```

<!-- ***** -->
<!-- ***** Classe Professor ***** -->
<!-- ***** -->
<owl:Class rdf:about="Professor">
  <!-- Atributs amb valor -->
  <rdfs:subClassOf rdf:resource="Persona"/>
  <owl:Restriction>
    <owl:onProperty rdf:resource="sap"/>
    <owl:qualifiedCardinality
rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    <owl:onDataRange rdf:resource="&xsd:string"/>
  </owl:Restriction>
  <owl:Restriction>
    <owl:onProperty rdf:resource="especialitat"/>
    <owl:qualifiedCardinality
rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    <owl:onDataRange rdf:resource="&xsd:string"/>
  </owl:Restriction>

  <!-- Atribut que referencia altres classes -->
  <owl:Restriction>
    <owl:onProperty rdf:resource="ensenya"/>
    <owl:someValuesFrom rdf:resource="Alumne"/>
  </owl:Restriction>
</owl:Class>

```

### Exemple de creació d'instàncies

```

<!-- ***** -->
<!-- ***** Classe Professor ***** -->
<!-- ***** -->
<owl:NamedIndividual rdf:about="professorXavi">
  <rdf:type rdf:resource="Professor"/>
  <nom rdf:datatype="&xsd:string">Xavi</nom>
  <sap rdf:datatype="&xsd:string">Música</sap>
  <especialitat rdf:datatype="&xsd:string">piano</especialitat>
  <ensenya rdf:resource="alumneRamon"/>
  <ensenya rdf:resource="alumneMaribel"/>
  <ensenya rdf:resource="alumneJordi"/>
</owl:NamedIndividual>

```

## XQuery

*XQuery* és un llenguatge per fer consultes sobre documents *XML*. El resultat de la cerca es pot convertir fàcilment en un altre document *XML*, *HTML* o qualsevol altre tipus de document de text.

Una consulta *XQuery* pot estar formada per codi *Xpath* més cinc tipus de sentències diferents:

- **FOR:** Fa un recorregut sobre una llista d'elements i en cada iteració assigna el següent valor a una variable.
- **LET:** Assigna tota la llista d'elements a una variable.
- **WHERE:** Imposa la condició que han de complir els elements que formin part del resultat.
- **ORDER BY:** Ordena els elements resultants.
- **RETURN:** Genera la sortida.

Tenint el document *XML*:

```
<llistapersones>
  <persona>
    <nom>Marta</nom>
    <edad>45</edad>
  </persona>
  <persona>
    <nom>Pau</nom>
    <edad>12</edad>
  </persona>
  <persona>
    <nom>Mireia</nom>
    <edad>17</edad>
  </persona>
  <persona>
    <nom>Raül</nom>
    <edad>23</edad>
  </persona>
</llistapersones>
```

Es podria obtenir una llista de les persones majors d'edat amb la següent consulta:

```
<majorsedad>{
  for $per in doc("persones.xml")/llistapersones/persona
  where $per/edad >= 18
  return <persona>{data($per/nom)}</persona>
}</majorsedad>
```

I fent servir sintaxi *Xpath*:

```
<majorsedad>{
  for $per in doc("persones.xml")/llistapersones/persona[./edad>=18]
  return <persona>{data($per/nom)}</persona>
}</majorsedad>
```

## Java-Android

---

Java és el llenguatge més habitual per programar aplicacions Android. La màquina virtual d'Android, la *Dalvik*, fa servir una bona part de l'API estàndard de Java i afegeix un API propi per poder gestionar la interacció amb els diferents elements del terminal (físics o lògics) com: el GPS, la càmera de fotos, l'agenda del telèfon, la recepció d'SMS...

Una aplicació Android pot tenir fins a quatre tipus de components diferents:

- **Activities:** Són els més habituals. Cada pantalla d'un programa acostuma a estar associat a un *activity* que s'encarrega de gestionar-la.
- **Services:** Un *service* és un component que s'executa en segon pla i no mostra cap pantalla. Es poden fer servir aquest tipus de serveis, per exemple, en un programa per escoltar música, no té sentit que mentre s'estigui reproduint una cançó no es pugui fer res més amb el mòbil.
- **Content providers:** Els *content provider* s'encarreguen de controlar un conjunt de dades i oferir-les a la resta d'aplicacions del dispositiu. Un exemple de *content provider* seria l'agenda de contactes.
- **Broadcast receivers:** És un component que s'encarrega d'escoltar i donar resposta als anuncis del sistema, per exemple que la bateria s'està esgotant o que algú està trucant per telèfon...

Quan es crea un programa Android s'ha de generar un fitxer *AndroidManifest.xml* on s'han de registrar els recursos que es necessitaran fer servir (accés a la agenda de contactes, accés a Internet, llegir o escriure a la targeta de memòria...) per tal de que l'usuari pugui decidir fer la instal·lació o no. Altres elements que també s'han de registrar al manifest són el components que s'executaran (activities, services...).

També es pot especificar, al manifest, la versió mínima de l'API on es podrà executar l'aplicació, la resolució de pantalla a la que va dirigida, etc. Tot això servirà perquè quan es faci una cerca al *Market*<sup>2</sup>, des d'un terminal, s'eliminin tots aquells resultats que no podrien executar-se correctament per no ser compatibles amb el dispositiu.

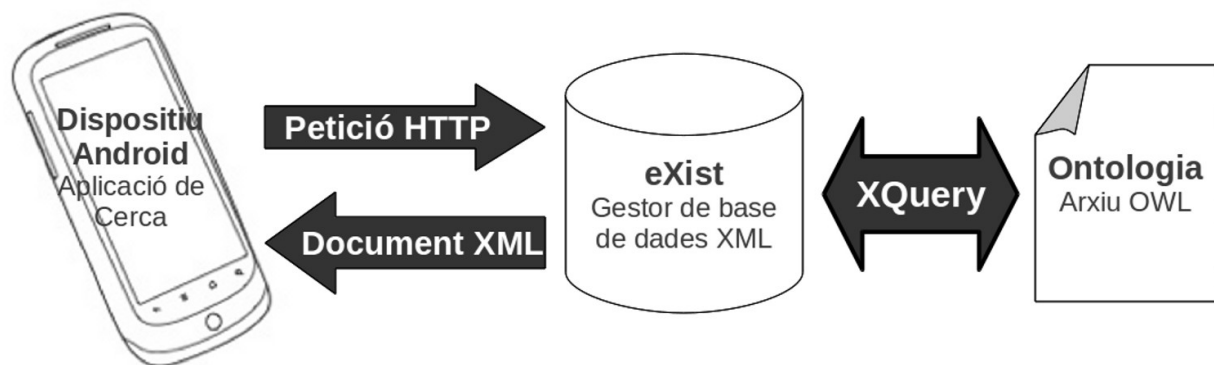
---

<sup>2</sup> Botiga online de google d'on es poden descarregar i instal·lar aplicacions Android.

## Una visió a gran escala

Abans d'aprofundir en cadascuna de les parts que constituïran el projecte *petraCAT*, pot ser interessant tenir una visió global de com s'uniran i interactuaran les diferents peces.

Els quatre grans blocs que es poden diferenciar a primera vista són: l'ontologia, el gestor de base de dades, els documents XML i el programa de cerca.



### *Interacció dels diferents components del projecte*

L'ontologia serà qui tindrà tota la informació de les instàncies (vies d'escalada, refugis, zones...) i de com es relacionen entre elles. Aquesta ontologia estarà emmagatzemada dins del gestor de base de dades *eXist*.

El gestor *eXist*, a més de l'ontologia, tindrà emmagatzemades una serie de consultes xQuery que generen documents XML.

El programa de cerca, que estarà instal·lat en un dispositiu mòbil amb connexió a Internet, cada cop que tingui necessitat de dades, llençarà una petició de cerca via HTTP contra *eXist*. En aquest moment s'executarà una de les consultes xQuery, segons la petició feta per l'aplicació, i s'enviarà al dispositiu mòbil el document XML generat.

# L'ontologia

Aquesta ontologia pretén estructurar el coneixement que hi ha al voltant d'una via d'escalada en roca. Queden fora d'aquesta ontologia altres tipus de vies (ferrades, corredors, cascades de gel...) que podrien formar part d'una ampliació futura.

## Jerarquia de classes

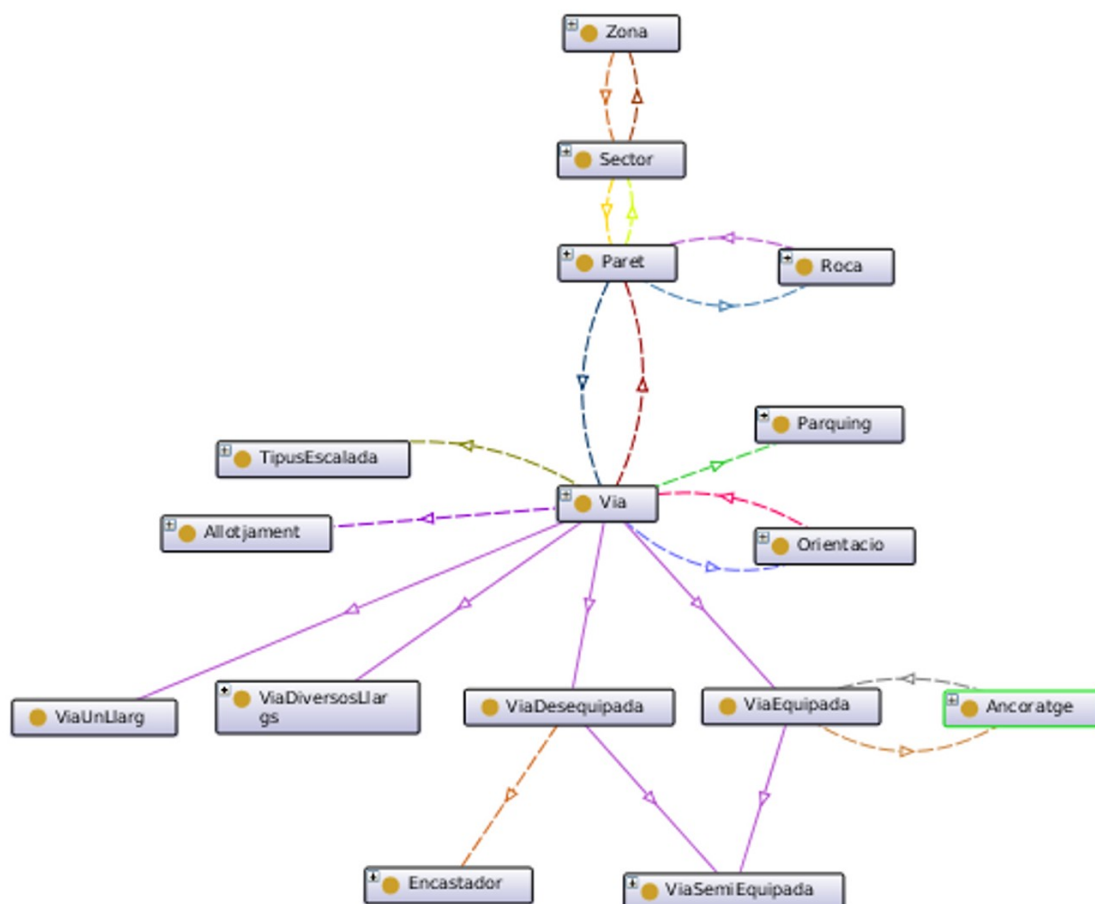
Del diagrama de la jerarquia de classes presentat en aquest apartat, s'hauria de destacar el fet que una via pot ser de un o diversos llargs i, a la vegada, pot ser equipada, desequipada o semiequipada.



Diagrama de classes de l'ontologia

## Relacions entre classes

A continuació es presenta un diagrama amb les relacions principals entre classes. Es fàcil de veure que tot gira al voltant de la *classe Via* (una via d'escalada en roca) l'element més important en aquest treball.



*Diagrama de relacions entre classes de l'ontologia*

Tot i que el programa només farà cerques en un sentit, s'han volgut contemplar les relacions inverses perquè l'ontologia hauria de tenir sentit per sí sola i s'hauria de poder fer servir per fer consultes, sobre l'àrea de coneixement que representa, molt més enllà del que pugui fer la petita aplicació de mostra.



## Atributs de les classes

---

La major part de les classes tenen un únic atribut per saber el nom de la instància concreta, per exemple: el nom d'una zona d'escalada, el nom d'un tipus d'ancoratge, el nom d'un tipus de roca...

Totes aquestes classes amb un únic atribut seran útils per definir diferents conjunts de vies d'escalada organitzades d'alguna manera concreta, per exemple: el conjunt de totes les vies orientades al sud, el conjunt de totes les vies de la zona de Montserrat, el conjunt de totes les vies que s'escalen en artificial... Fent la intersecció d'aquest conjunts podríem obtenir el conjunt de totes les vies de la zona de Montserrat que s'escalen en artificial i que estan orientades al sud.

La resta d'atributs estaran situats sobre els *punts d'interès* i les seves subclasses. De cada *punt d'interès* voldrem saber la seva *latitud* i la seva *longitud*. Aquestes informacions s'ampliaran, en el cas dels *al·lotjaments*, amb *l'adreça*, la *població*, el *telèfon* de contacte... En el cas de les vies d'escalada, amb els *metres* que té i, si és de més d'un llarg, quants són aquests *llargs*, etc.

## Els documents

Tenint en compte que les connexions per a dispositius mòbils encara són cares i limiten la quantitat de descàrrega a màxima velocitat, s'han dissenyat dos documents XML diferents per intentar reduir al mínim la transferència d'informació innecessària. El primer document contindrà la llista de vies que compleixen els requisits de la cerca i el segon, la informació referent a una via. D'aquesta manera no caldrà descarregar la informació de totes les vies si no les hem de consultar.

### Llista de vies

Aquest document contindrà la llista de vies, que han passat el filtre de cerca, agrupades per *zona* i *sector*. Sobre les vies només es donarà alguna informació bàsica: els *metres* que té, la *dificultat màxima* i la *dificultat màxima obligatòria*.

### XML Schema

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <!-- ***** -->
  <!-- ***** tipus simples ***** -->
  <!-- ***** -->
  <!-- tipus metrestype -->
  <xs:simpleType name="metrestype">
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="0" />
    </xs:restriction>
  </xs:simpleType>

  <!-- ***** -->
  <!-- ***** tipus complexos ***** -->
  <!-- ***** -->
  <!-- tipus viatype -->
  <xs:complexType name="viatype">
    <xs:sequence>
      <xs:element name="nom" type="xs:string" />
      <xs:element name="dmaxima" type="xs:string" />
      <xs:element name="dobligada" type="xs:string" />
      <xs:element name="metres" type="metrestype" />
    </xs:sequence>
  </xs:complexType>

  <!-- tipus parettype -->
  <xs:complexType name="parettype">
    <xs:sequence>
      <xs:element name="via" type="viatype" />
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

```

        </xs:sequence>
        <xs:attribute name="nom" type="xs:string" use="required" />
</xs:complexType>

<!-- tipus sectortype -->
<xs:complexType name="sectortype">
  <xs:sequence>
    <xs:element name="paret" type="parettype" />
  </xs:sequence>
  <xs:attribute name="nom" type="xs:string" use="required" />
</xs:complexType>

<!-- tipus resumtype -->
<xs:complexType name="resumtype">
  <xs:sequence>
    <xs:element name="zona" type="zonatype" />
  </xs:sequence>
</xs:complexType>

<!-- tipus zonatype -->
<xs:complexType name="zonatype">
  <xs:sequence>
    <xs:element name="sector" type="sectortype" />
  </xs:sequence>
  <xs:attribute name="nom" type="xs:string" use="required" />
</xs:complexType>

<!-- ***** -->
<!-- ***** element principal ***** -->
<!-- ***** -->
  <xs:element name="resum" type="resumtype" />
</xs:schema>

```

## Exemple de document

```

<resum>
  <zona nom="Castellfollit de Poblet">
    <sector nom="Pic de l'Àliga">
      <paret nom="Pic de l'Àliga">
        <via>
          <nom>Aresta GER</nom>
          <dmaxima>V</dmaxima>
          <dobligada>V</dobligada>
          <metres>180</metres>
        </via>
      </paret>
    </sector>
  </zona>
  <zona nom="La Mussara">
    <sector nom="Antenes">
      <paret nom="Covallonga">
        <via>

```

```

        <nom>Covallonga</nom>
        <dmaxima>V</dmaxima>
        <dobligada>V</dobligada>
        <metres>45</metres>
    </via>
</paret>
</sector>
</zona>
</resum>

```

## Informació d'una via

Aquest document donarà tota la informació que es té d'una via: dades tècniques, esquema del recorregut, on aparcar, com arribar i com tornar, allotjaments propers...

## XML Schema

```

<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

    <!-- ***** -->
    <!-- ***** tipus simples ***** -->
    <!-- ***** -->
    <!-- tipus naturaltype -->
    <xs:simpleType name="naturaltype">
        <xs:restriction base="xs:integer">
            <xs:minInclusive value="0" />
        </xs:restriction>
    </xs:simpleType>

    <!-- ***** -->
    <!-- ***** tipus complexos ***** -->
    <!-- ***** -->
    <!-- tipus gpstype -->
    <xs:complexType name="gpstype">
        <xs:sequence>
            <xs:element name="latitud" type="double" />
            <xs:element name="longitud" type="double" />
        </xs:sequence>
    </xs:complexType>

    <!-- tipus equipamenttype -->
    <xs:complexType name="equipamenttype">
        <xs:sequence>
            <xs:element name="anclatge" type="xs:string" />
        </xs:sequence>
    </xs:complexType>

    <!-- tipus materialtype -->
    <xs:complexType name="materialtype">

```

```

    <xs:sequence>
      <xs:element name="mat" type="xs:string" />
    </xs:sequence>
  </xs:complexType>

  <!-- tipus parkingtype -->
  <xs:complexType name="parkingtype">
    <xs:sequence>
      <xs:element name="gps" type="gpstype" />
      <xs:element name="acces" type="xs:string" />
      <xs:element name="descripcio" type="xs:string" />
    </xs:sequence>
  </xs:complexType>

  <!-- tipus enllacostype -->
  <xs:complexType name="enllacostype">
    <xs:sequence>
      <xs:element name="enllac" type="xs:anyURI" />
    </xs:sequence>
  </xs:complexType>

  <!-- tipus serveistype -->
  <xs:complexType name="serveistype">
    <xs:sequence>
      <xs:element name="guardat" type="xs:boolean" />
      <xs:element name="places" type="naturaltype" />
      <xs:element name="placesng" type="naturaltype" />
      <xs:element name="llum" type="xs:boolean" />
      <xs:element name="aigua" type="xs:boolean" />
      <xs:element name="calefaccio" type="xs:boolean" />
      <xs:element name="lliteres" type="xs:boolean" />
      <xs:element name="matalassos" type="xs:boolean" />
      <xs:element name="wc" type="xs:boolean" />
      <xs:element name="cuina" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>

  <!-- tipus contactetype -->
  <xs:complexType name="contactetype">
    <xs:sequence>
      <xs:element name="telefon" type="xs:string" />
      <xs:element name="mail" type="xs:string" />
    </xs:sequence>
  </xs:complexType>

  <!-- tipus allotjamenttype -->
  <xs:complexType name="allotjamenttype">
    <xs:sequence>
      <xs:element name="gps" type="gpstype" />
      <xs:element name="acces" type="xs:string" />
      <xs:element name="adreca" type="xs:string" />
      <xs:element name="poblacio" type="xs:string" />
    </xs:sequence>
  </xs:complexType>

```

```

        <xs:element name="serveis" type="serveistype" />
        <xs:element name="contacte" type="contactetype" />
    </xs:sequence>
    <xs:attribute name="tipus" type="xs:string" use="required" />
</xs:complexType>

<!-- tipus allotjamentstype -->
<xs:complexType name="allotjamentstype">
    <xs:sequence>
        <xs:element name="allotjament" type="allotjamenttype" />
    </xs:sequence>
</xs:complexType>

<!-- tipus viatype -->
<xs:complexType name="viatype">
    <xs:sequence>
        <xs:element name="nom" type="xs:string" />
        <xs:element name="paret" type="xs:string" />
        <xs:element name="zona" type="xs:string" />
        <xs:element name="sector" type="xs:string" />
        <xs:element name="gps" type="gpstype" />
        <xs:element name="dificultatM" type="xs:string" />
        <xs:element name="dificultatO" type="xs:string" />
        <xs:element name="metres" type="xs:string" />
        <xs:element name="llargs" type="xs:string" />
        <xs:element name="escalada" type="xs:string" />
        <xs:element name="tipus" type="xs:string" />
        <xs:element name="equipament" type="equipamenttype" />
        <xs:element name="orientacio" type="xs:string" />
        <xs:element name="roca" type="xs:string" />
        <xs:element name="material" type="materialtype" />
        <xs:element name="esquema" type="xs:anyURI" />
        <xs:element name="parquing" type="parquingtype" />
        <xs:element name="aproximacio" type="xs:string" />
        <xs:element name="descens" type="xs:string" />
        <xs:element name="allotjaments"
type="allotjamentstype" />
        <xs:element name="enllacos" type="enllacostype" />
    </xs:sequence>
</xs:complexType>

<!-- ***** -->
<!-- ***** element principal ***** -->
<!-- ***** -->
<xs:element name="via" type="viatype" />

</xs:schema>

```

**Exemple de document**

```

<via>
  <zona>La Mussara</zona>
  <sector>Antenes</sector>
  <paret>Covallonga</paret>
  <nom>Covallonga</nom>
  <gps>
    <latitud>41.244666</latitud>
    <longitud>1.055677</longitud>
  </gps>
  <dificultatM>V</dificultatM>
  <dificultatO>V</dificultatO>
  <metres>45</metres>
  <llargs>2</llargs>
  <escalada>Lliure</escalada>
  <tipus>Equipada</tipus>
  <orientacio>Sud</orientacio>
  <roca>Calcaria</roca>
  <esquema>http://www.ressenya.net/mussara/covallonga.jpg</esquema>
  <aproximacio>Al final del parking trobarem una bassa, just per allí
passa el GR-172 que hem de seguir en direcció a les antenes (repetidor de
La Mussara). Quan trobem una petita font, abans de que el camí comenci a
agafar pendent, ens desviarem per un petit senderó que surt a l'esquerra
i que ens portarà a peu de via en uns 10 minuts.</aproximacio>
  <descens>Es pot baixar rapelant per la mateixa via o per una petita
ferrata que hi ha a la dreta de la via, encara que aquesta última opció
també ens obligarà a rapelar l'últim llarg de la via.</descens>
  <enllacos>
    <enllac>http://www.meteo.cat</enllac>
  </enllacos>
  <equipament>
    <ancoratge>Parabolts</ancoratge>
  </equipament>
  <parking>
    <gps>
      <latitud>41.243158</latitud>
      <longitud>1.061301</longitud>
    </gps>
    <acces>Des de Vilaplana en direcció Prades trobarem un Pàrquing a
ma esquerre cap al Km 15,5 aprox.</acces>
    <descripcio>Zona amb molt espai per aparcar i espai de pas per a
caminants del GR-172. Si estigués ple, a l'altre costat de la carretera
hi ha espai per molts més cotxes.</descripcio>
  </parking>
  <allotjaments>
    <allotjament tipus="refugi">
      <nom>La Mussara</nom>
      <gps>
        <latitud>1.3322445</latitud>
        <longitud>43.2233211</longitud>
      </gps>
      <acces>Seguir la carretera de Vilaplana a la Mussara. El
refugi està entrant per un camí a l'esquerra a 200 m del poble de la
Mussara.</acces>
      <poblacio>La Mussara</poblacio>
      <serveis>

```

```
<guardat>true</guardat>
<places>54</places>
<llum>true</llum>
<aigua>true</aigua>
<calefaccio>true</calefaccio>
<lliteres>true</lliteres>
<matalassos>true</matalassos>
<wc>true</wc>
<cuina>true</cuina>
</serveis>
<contacte>
  <telefon>629750484</telefon>
</contacte>
</allotjament>
</allotjaments>
</via>
```



## Les consultes xQuery

L'ontologia estarà emmagatzemada en format OWL dins d'una col·lecció *eXist*. Es podrà obtenir informació de l'ontologia fent servir consultes xQuery emmagatzemades al directori *webapp/xquery* de la instal·lació de *l'eXist*.

### Exemple de consulta

Com a exemple de consulta es mostrarà el codi xQuery que obté una llista de vies, cercant a tots els sectors existents a la base de dades, segons els paràmetres de cerca especificats (dificultat màxima, orientació, tipus de roca...)

```
xquery version "1.0";
declare namespace rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#";
declare namespace owl="http://www.w3.org/2002/07/owl#";
declare namespace petra =
"http://www.semanticweb.org/ontologies/2011/4/PetraCATontologi.owl#";
declare namespace request="http://exist-db.org/xquery/request";

let $orientacio := concat("#",request:get-parameter("orientacio", ""))
let $escalada := concat("#",request:get-parameter("escalada", ""))
let $tipus := concat("#",request:get-parameter("tipus", ""))
let $llargs := concat("#",request:get-parameter("llargs", ""))
let $roca := concat("#",request:get-parameter("roca", ""))
let $dmax := request:get-parameter("dmaxima", "")

return
<resum>{
  for $zona in
collection('petraCAT')//owl:NamedIndividual[./rdf:type/@rdf:resource="http://www.
semanticweb.org/ontologies/2011/4/PetraCATontologi.owl#Zona"]
  return
    <zona nom="{data($zona/petra:nom)}">{
      for $sector in
collection('petraCAT')//owl:NamedIndividual[./rdf:type/@rdf:resource="http://www.
semanticweb.org/ontologies/2011/4/PetraCATontologi.owl#Sector"]
        where data($zona/@rdf:about) =
data($sector/petra:isSectorOf/@rdf:resource)
        return
          <sector nom="{data($sector/petra:nom)}">{
            for $paret in
collection('petraCAT')//owl:NamedIndividual[./rdf:type/@rdf:resource="http://www.
semanticweb.org/ontologies/2011/4/PetraCATontologi.owl#Paret"]
              where data($sector/@rdf:about) =
data($paret/petra:isParetOf/@rdf:resource)
              return
                <paret nom="{data($paret/petra:nom)}">{
                  for $via in
collection('petraCAT')//owl:NamedIndividual[./rdf:type/@rdf:resource="http://www.
semanticweb.org/ontologies/2011/4/PetraCATontologi.owl#Via") and
(data($paret/@rdf:about) = data(/petra:isViaOf/@rdf:resource))
                    where (($orientacio="#" or ends-
with(data($via/petra:hasOrientacio/@rdf:resource), $orientacio)) and
($escalada="#" or ends-with(data($via/petra:isClimbedIn/@rdf:resource),
```

```

$escalada)) and($tipus="#" or ends-with(data($via/rdf:type/@rdf:resource),
$tipus)) and($llargs="#" or ends-with(data($via/rdf:type/@rdf:resource),$llargs))
and($roca="#" or ends-with(data($paret/petra:isMadeOfRoca/@rdf:resource), $roca))
and ($dmax="#" or
contains($dmax,concat(":",data($via/petra:dificultadMaxima),":")))
    return
    <via>
      <nom>{data($via/petra:nom)}</nom>
      <dmaxima>{data($via/petra:dificultadMaxima)}</dmaxima>
      <obligada>{data($via/petra:dificultadMaximaObligatoria)}<
/dobligada>
      <metres>{data($via/petra:metres)}</metres>
    </via>
  </paret>
}</sector>
}</zona>
}</resum>

```

## Crida a una consulta xQuery

La manera en que es cridaran les consultes xQuery emmagatzemades a *l'eXist* serà mitjançant peticions HTTP. Per exemple, si es volguessin cercar les vies orientades al sud, de més d'un llarg i que s'escalin en lliure a tots els sectors existents a la base de dades (que en aquest cas es troba a l'adreça 192.168.2.11) s'hauria de fer la següent petició HTTP:

```

http://192.168.2.11:8080/exist/xquery/viestotssectors.xql?
escalada=Lliure&orientacio=Sud&llargs=ViaDiversosLlargos

```

## L'aplicació

La informació i el coneixement, sense una manera de tractar-los i d'accedir fàcil i intuïtiva, no tenen gaire sentit. Aquest petit programa tractarà d'aportar una mica de sentit a la feina feta amb l'ontologia.

### Marc d'execució

---

L'aplicació ha de poder proporcionar informació sobre el terreny, per tant haurà de funcionar sobre un dispositiu mòbil amb connexió a Internet, com a mínim, i GPS, per poder fer servir tota la funcionalitat.

En aquest cas s'ha triat la plataforma Android pels següents motius:

- Proporciona un entorn d'execució prou potent per desenvolupar aquesta aplicació i les futures ampliacions, de les que es parlarà més endavant.
- La seva gran expansió. Al mercat es poden trobar una gran quantitat de dispositius mòbils Android accessibles a la majoria de butxaques. Quants més usuaris potencials hi hagin, més gent es podrà beneficiar d'aquest programa.
- La facilitat per fer arribar les aplicacions als usuaris finals, en contraposició a altres plataformes equivalents que només posen entrebancs.
- El fet de que es tracti d'un sistema operatiu lliure (almenys en la seva llicència) el fa més proper a la filosofia amb que s'ha volgut desenvolupar aquest treball.

### Una vista general

---

Comencem per veure les diferents pantalles de l'aplicació i com encaixen entre elles.

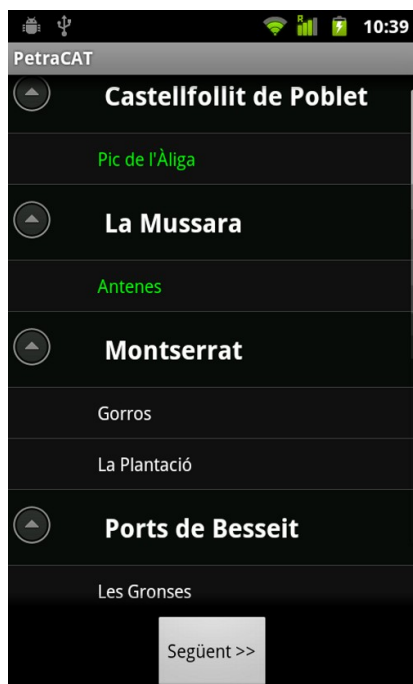
Es podrà cercar una via d'escalada de tres maneres diferents: Accedint a tot el contingut de la base de dades, accedint al contingut d'alguns sectors de diferents zones d'escalada i segons un àrea circular amb centre a les coordenades GPS actuals. A totes aquestes opcions s'accedirà des de la pantalla principal del programa.



*Aplicació - Pantalla principal*

Si es tria l'opció *Segons Zona/Sector*, s'anirà a parar a un altra pantalla on es podran escollir, marcant les caselles corresponents, sobre quines zones i quins sectors d'aquestes zones es vol fer la cerca.

En el cas que l'opció triada sigui *Segons Posició GPS*, apareixerà una pantalla amb les coordenades GPS actuals i una capsa de text per donar el diàmetre de l'àrea de cerca en metres.



*Aplicació - Segons Zona/Sector*



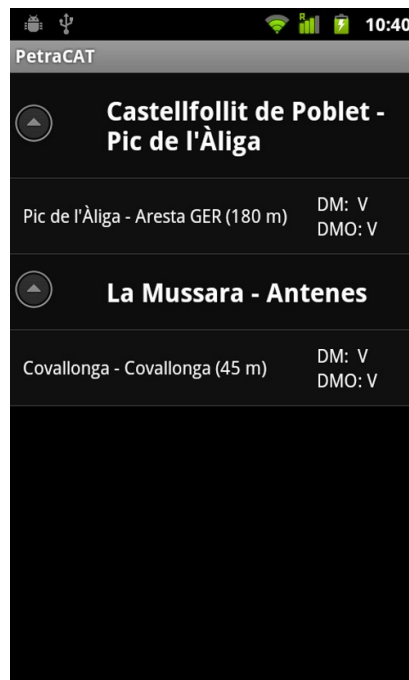
*Aplicació- Segons posició GPS*

Un cop decidida l'opció de cerca, tan si s'ha passat per una de les pantalles anteriors com si no, es presentarà un formulari on podrem definir les característiques de la via que s'està buscant.



*Aplicació - Formulari de cerca*

El resultat de la cerca es veurà en forma de llista agrupada per zona i sector. La llista també mostrarà algunes informacions sobre les vies: els metres que té, la DM (dificultat màxima) i la DMO (dificultat màxima obligatòria).

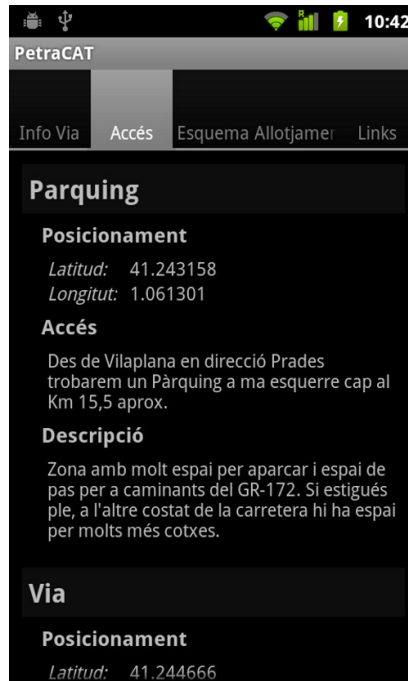


*Aplicació - Resultat de la cerca*

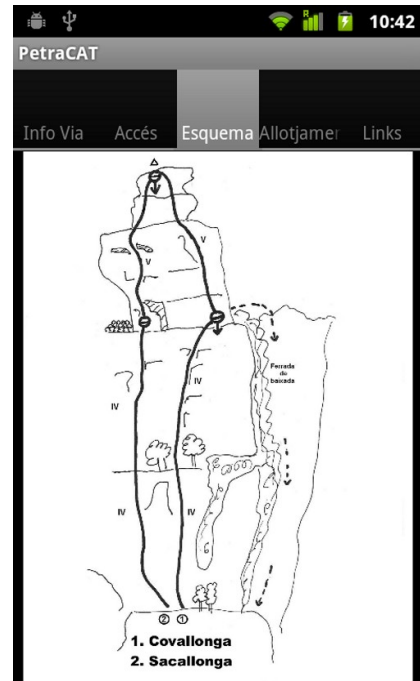
Seleccionant una de les vies es podrà accedir a tota la seva informació: Les dades tècniques, un esquema del seu recorregut, els allotjaments propers (refugis, albergs...) i una serie d'enllaços a webs, blogs... amb més informació sobre la via triada.



Aplicació - Via - Informació tècnica



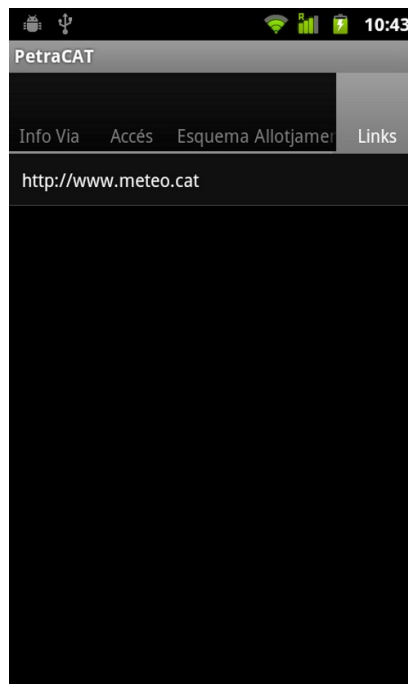
Aplicació - Via - Accés



Aplicació - Via - Esquema

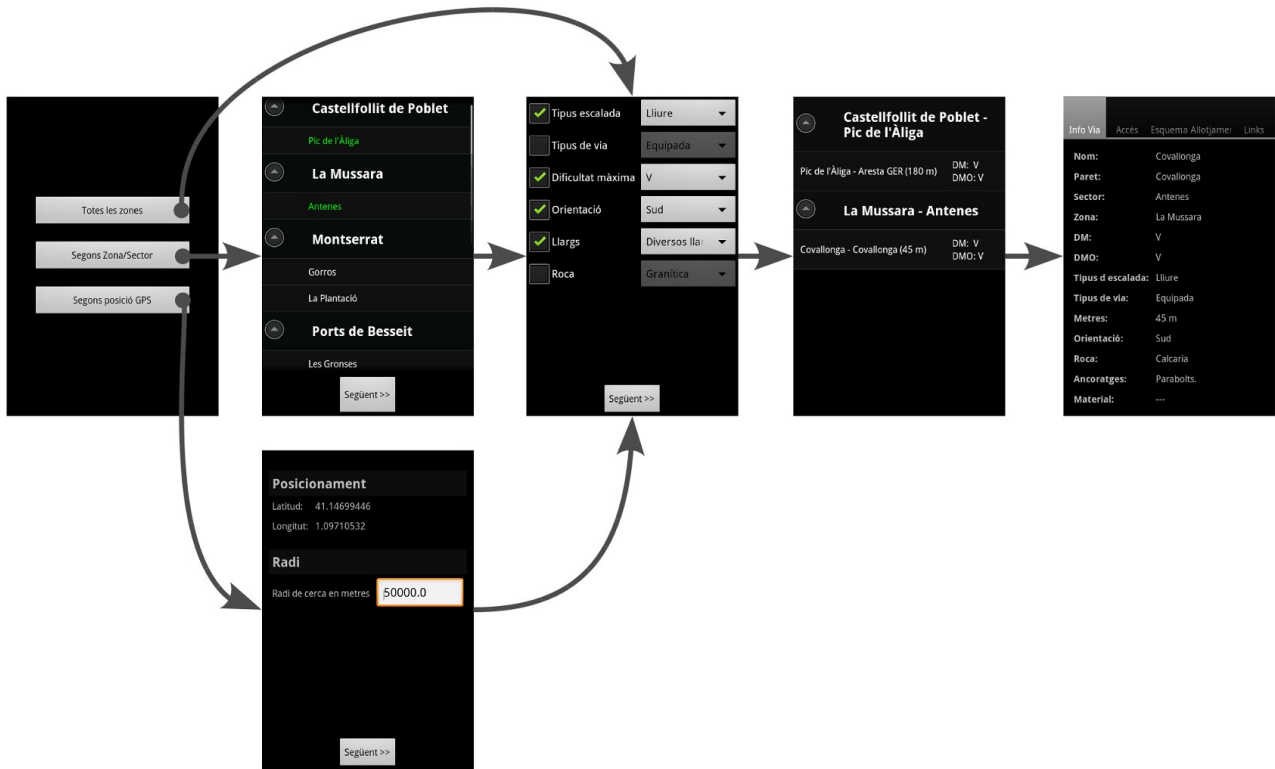


Aplicació - Via - Allotjaments



Aplicació - Via - Links

Fent servir el *botó de menú*, dels terminals, s'accedeix a un menú general d'on es pot iniciar una nova consulta o mostrar la pantalla d'informació. A més, des de qualsevol pantalla es pot tornar a l'anterior, fent servir el *botó per tirar enrere* que porten tots els terminals, per si cal fer algun canvi.



Esquema de transició entre pantalles

Es tracta d'una aplicació molt simple i intuïtiva que dona un accés fàcil a la informació tècnica de la via seleccionada però que, a més, permet accedir a molta més informació per poder planificant l'escalada del dia següent des del refugi o per si s'ha de fer un canvi de plans per causes inesperades i traslladar-se a un altra zona o sector.

## Diagrama de classes

Al diagrama es mostren les classes principals de l'aplicació i les seves relacions. S'ha seguit el patró MVC (Model Vista Control) tan a nivell general com a més petita escala, per exemple en el tractament de les dades que fan les llistes, ja que Android fa servir aquest patró per treballar amb alguns dels seus components gràfics. Per simplificar el diagrama s'han obviat la majoria de les classes que representen objectes gràfics (Button, Spinner, CheckBox...) per no aportar informació útil, més aviat el contrari, el fan més enrevessat.

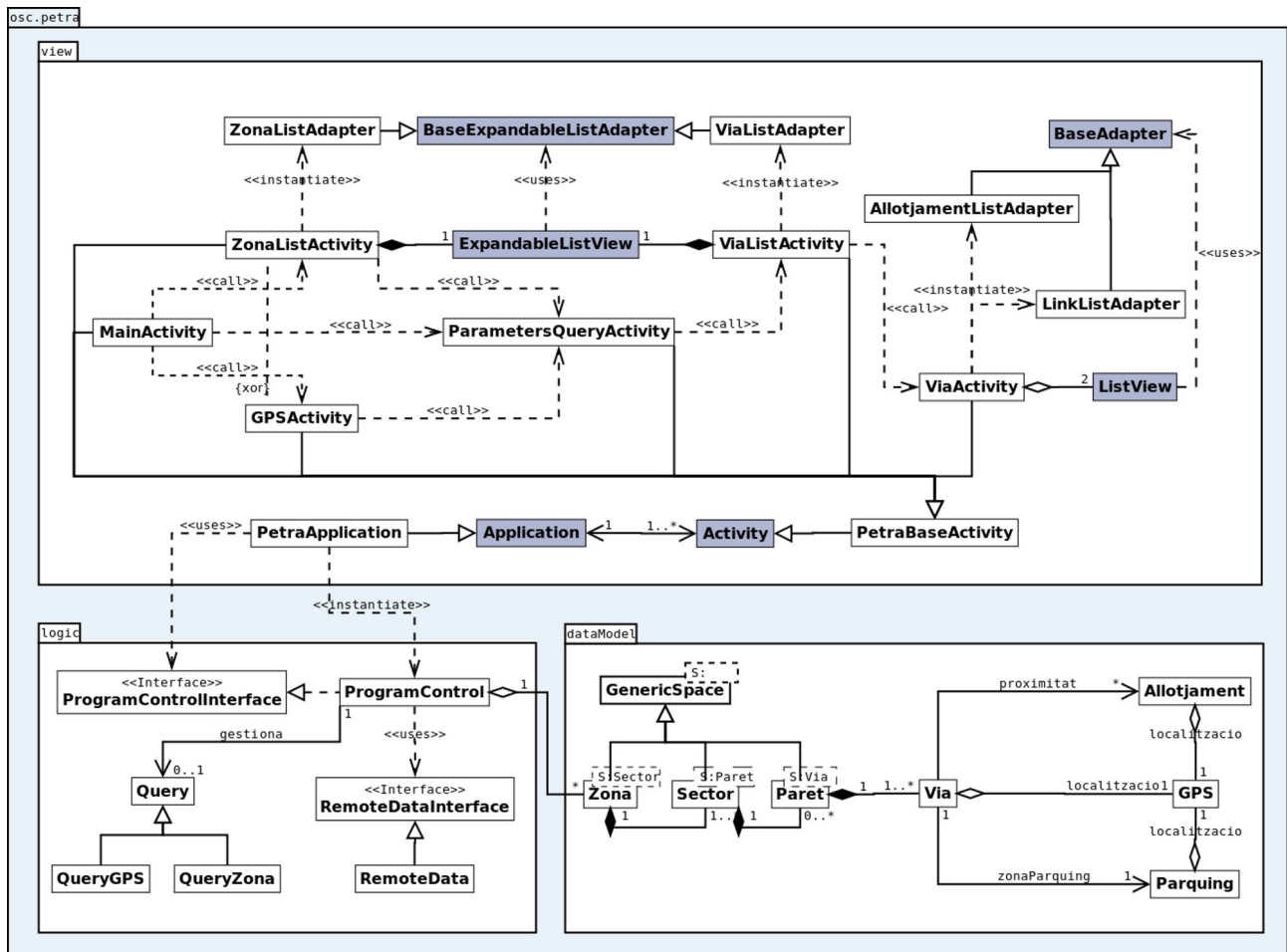


Diagrama de classes

Per veure una versió més completa del diagrama de classes es pot anar al pòster que hi ha al final del document.



## Diagrama de seqüències (seguiment d'una cerca)

La millor manera de veure el funcionament intern de la cerca de vies d'escalada en aquesta aplicació és amb un diagrama de seqüències.

Com en el cas del diagrama de classes, s'ha intentat simplificar-ho al màxim fent només les crides necessàries per entendre la interacció entre els objectes. S'han omès completament els *Intents*<sup>3</sup>, les classes anònimes i la major part dels elements gràfics.

Pel mateix motiu, la simplicitat, s'ha suposat que, en aquest exemple, la cerca només es farà per tipus de roca a qualsevol zona i sector de Catalunya.

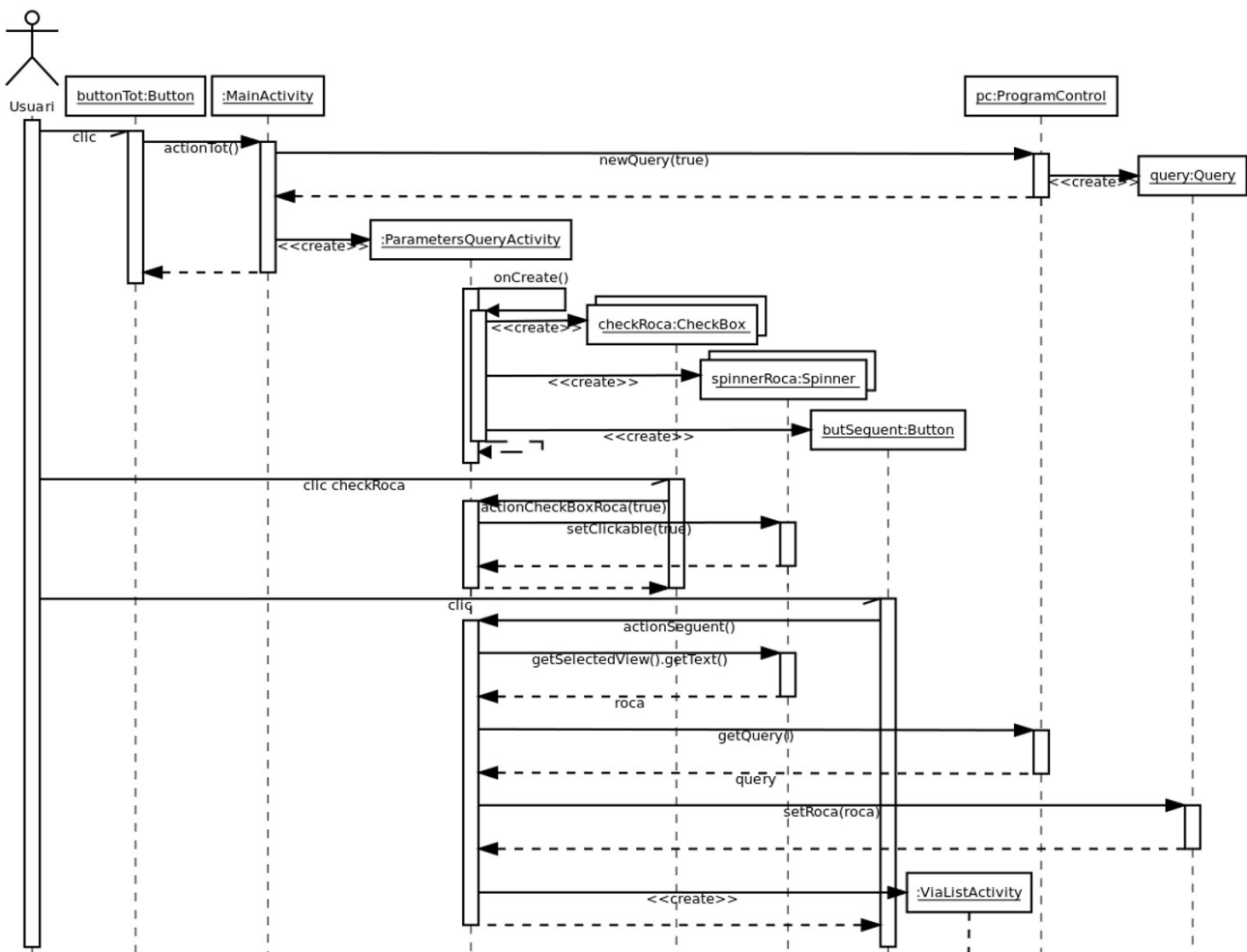


Diagrama de seqüències (primera part). Cerca, a tots els sectors i zones de Catalunya, de vies d'escalada segons el tipus de roca.

<sup>3</sup> Classe que permet executar un altre Activity o fer una crida a un programa o servei extern.

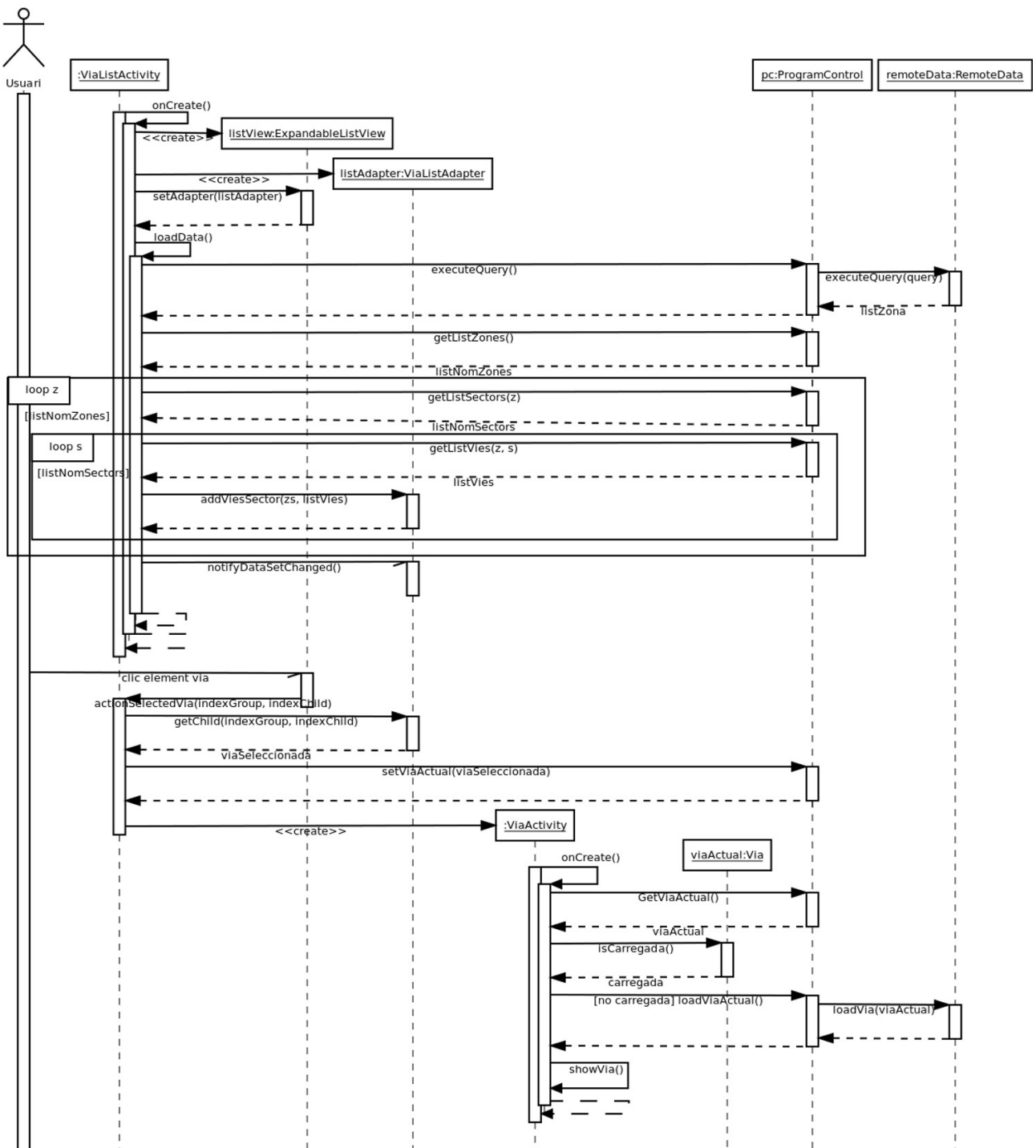


Diagrama de seqüències (segona part). Cerca, a tots els sectors i zones de Catalunya, de vies d'escalada segons el tipus de roca.

## Línies de continuació del projecte

Aquest és encara un projecte molt viu i amb moltes possibilitats d'ampliació. Algunes d'aquestes possibilitats són:

- **Ampliar l'ontologia amb més tipus de vies (corredors de neu, cascades de gel, crestes, ferrades...).** Això també obligaria a adaptar una mica l'aplicació als nous tipus de via.
- **Expandir l'ontologia per poder emmagatzemar vies de tot el mon.** Igual que en el cas anterior s'hauria d'adaptar l'aplicació per acotar les cerques a un país o regió triats i s'hauria de traduir a les llengües més utilitzades.
- **Afegir mapes a l'aplicació.** Permetria fer cerques segons una posició seleccionada al mapa, permetria visualitzar tots els punts d'interès dins d'un mapa i, inclús, establir rutes entre els punts d'interès o entre la posició GPS actual i un punt d'interès.
- **Generar documents pdf de la informació obtinguda sobre una via.** D'aquesta manera es podria compartir la informació trobada amb la resta de membres del grup d'una manera més còmoda.
- **Substituir el gestor de base de dades eXist per virtuosoDB.** Proporciona un marc de treball més adient per tractar amb ontologies.
- **Transformar les consultes XQuery en consultes SPARQL.** XQuery és un llenguatge per fer consultes sobre documents XML, això vol dir que no té en compte totes les característiques que té una ontologia. Sense tenir gaire coneixement, SPARQL sembla més adient per fer consultes sobre ontologies.
- **Fer una versió del programa que pugui funcionar en navegadors.** Serà molt més còmode de fer servir quan es tingui un ordinador a mà i obrirà la possibilitat de fer consultes des de dispositius no Android.
- **Donar la possibilitat als usuaris d'introduir informació de noves vies.** Degut a la gran quantitat de vies existents, les úniques manera de fer créixer el suficient la base de dades són: (1) comptar amb els recursos només a l'abast de grans empreses o (2) confiar en els propis usuaris que amb poc esforç podrien aportar una gran quantitat d'informació. En aquest segon cas, s'hauria d'estudiar la forma de fer-ho per evitar la introducció de dades falses, per part d'usuaris mal intencionats, que podrien arribar a provocar accidents mortals.
- **Poder emmagatzemar localment vies cercades.** Fer servir aquesta aplicació mòbil en zones sense cobertura serà un fet normal. Tenir les vies emmagatzemades localment permetrà continuar treballant amb el programa, encara que només sigui amb un grup reduït de vies. Per un altra banda evitaria que s'hagués de tornar a buscar una via ja

trobada anteriorment i el cost extra de tornar a transmetre informació ja transmesa.

- **Modelar a nivell d'ontologia el coneixement sobre els graus de dificultat de les vies.** Fins ara l'ontologia només guarda aquesta informació com un atribut textual dins de la via. El tractament es fa a nivell d'aplicació i només tenint en compte un sistema que barreja la taula de dificultats de la UIAA i la francesa però que no es ni una ni l'altra (el sistema habitual a les nostres terres). Seria interessant poder fer servir qualsevol sistema de graduació<sup>4</sup> i establir equivalències entre elles a nivell d'ontologia.

---

<sup>4</sup> [http://es.wikipedia.org/wiki/Graduaci%C3%B3n\\_de\\_dificultad](http://es.wikipedia.org/wiki/Graduaci%C3%B3n_de_dificultad)

## Conclusions

Després del treball fet en aquest projecte s'ha pogut observar que:

- La web ha evolucionat molt ràpid des dels seus inicis, i la tendència és anar cap a la web semàntic, encara que només s'està començant i queda molta feina a fer.
- El futur més immediat de la web passa per les tecnologies basades en XML. OWL i RDF per modelar ontologies, Xpath per navegar per documents XML, Xquery per fer cerques d'informació dins de documents XML, SPARQL per fer consultes sobre RDF...
- Cada cop es fan servir dispositius mòbils més potents. Les connexions a Internet per aquest dispositius, encara que molt poc a poc, van reduint els seus preus i incrementant la seva velocitat. Com a conseqüència cada cop hi hauran més usuaris que accediran a Internet (per consultar el correu electrònic, accedir a xarxes socials, llegir blogs...) des dels dispositius mòbils.

Amb tot això, es pot concloure que: (1) no ha sigut la millor idea fer servir XQuery per fer les consultes, hauria sigut molt millor fer servir SPARQL, encara que funciona i es possible fer-ho; i (2) tot i que no s'hauria de descuidar la versió per a PC, ha sigut una bona decisió fer l'aplicació per a una plataforma mòbil.

## Glossari

- **Ancoratge fix:** Element fixat a la paret que es fa servir per passar la corda de l'escalador (fent servir una cinta exprés) i així evitar una caiguda mortal.
- **Buril:** Un dels primers ancoratges fixos que es va fer servir. Prové del món de la construcció. Actualment ja no s'instal·len, això vol dir que els que queden no són gaire fiables.
- **Cinta exprés:** Cinta molt resistent amb un mosquetó a cada extrem que es fa servir per unir la corda que porta l'escalador amb els ancoratges de la paret o els encastadors.
- **Clau:** Ancoratge que s'encasta a cops de martell a les fissures de la paret.
- **Encastador mecànic:** Encastador que té un mecanisme de lleves i molles que facilita molt la introducció a les fissures i permet que s'adapti molt millor a la forma que tingui. Actualment els encastadors mecànics més utilitzats són els friends.
- **Encastador passiu:** Tascó que es col·loca a les fissures de la paret, que té un petit cable d'acer d'on s'acostuma a penjar una cinta exprés per assegurar la progressió de l'escalador. Actualment hi han encastadors passius amb molts tipus de formes i noms, busquen adaptar-se el millor possible a la forma de la fissura.
- **Escalada artificial:** Forma d'escalada on l'escalador fa servir elements artificials, com pedals o les mateixes cintes exprés per on es passa la corda, per progressar. Es un tipus d'escalada on aconseguir un grau de dificultat més elevat no vol dir que la progressió sigui més difícil, si no més perillosa perquè els elements de seguretat es fan menys fiables. A les vies més extremes una caiguda significaria la mort.
- **Escalada lliure:** Forma d'escalada on l'escalador a de progressar fent servir només el seu cos, normalment peus i mans. La resta d'elements com les cordes, les cintes exprés, els encastadors... només els pot fer servir per protegir-se d'una caiguda, mai per progressar.
- **Espit:** Va ser el primer ancoratge fix d'expansió creat per escalada. Actualment encara es fan servir, encara que ja no és habitual instal·lar-los a les vies d'escalada. Aquest tipus d'ancoratge s'instal·la fent un forat a la roca per mitjans manuals.
- **Llarg:** Recorregut entre dues reunions.
- **Parabolt:** És l'ancoratge fix que més abunda a les nostres parets. És de tipus expansiu, com l'espit, però amb la diferència que en aquest cas s'ha de fer servir algun tipus de perforadora mecànica, a mà seria impossible fer el forat que es necessita.
- **Químic:** L'última generació d'ancoratges fixos i també la més resistent i fiable. Un ancoratge d'aquest tipus té forma d'espàrrec amb l'extrem exterior doblegat formant un

cercle. La instal·lació es fa perforant la roca i fent servir unes resines especials per fixar l'ancoratge a la paret.

- **Reunió:** Muntatge dins d'una paret pensat perquè sigui especialment resistent. És el punt on es troben els escaladors quan s'ha acabat una tirada de corda.
- **Via desequipada:** Via sense cap tipus d'ancoratge fix que protegeixi una possible caiguda de l'escalador. L'escalador ha d'autoprotegir-se fent servir els encastadors.
- **Via equipada:** Via d'escalada on cada pocs metres hi ha algun tipus d'ancoratge fix per protegir una possible caiguda de l'escalador.
- **Via semiequipada:** Via on només s'han equipat els passos més difícils i les reunions. S'han de fer servir encastadors per protegir-se d'una possible caiguda.

## Bibliografia

**F. Noy, Natalya i L. McGuinness, Deborah** (2005, 19 de setembre). *Desarrollo de Ontologías-101: Guía Para Crear Tu Primera Ontología* [document electrònic]. Stanford University.  
<[http://protege.stanford.edu/publications/ontology\\_development/ontology101-es.pdf](http://protege.stanford.edu/publications/ontology_development/ontology101-es.pdf)>

**L.Murphy, Mark** (2010). *Beginning Android 2* (1a Edició). USA: Apress

**Walmsley, Priscilla** (2007). *XQuery* (1a Edició). USA: O'REILLY

**Contreras, Jesús**. *Tutorial Ontologías* [document electrònic]  
<[http://www.sedic.es/gt\\_normalizacion\\_tutorial\\_ontologias.pdf](http://www.sedic.es/gt_normalizacion_tutorial_ontologias.pdf)>

**Horridge, Matthew** (2011, 24 de març). *A Practical Guide To Build OWL Ontologies Using Protégé 4 and CO-ODE Tools* (Edició 1.3) [document electrònic]. The University of Manchester.  
<[http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/resources/ProtegeOWLTutorialP4\\_v1\\_3.pdf](http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/resources/ProtegeOWLTutorialP4_v1_3.pdf)>

**Long, John i Gaines, Bob** (2006). *Anclajes de escalada REUNIONES Y SEGUROS* (2a Edició). Espanya: Ediciones Desnivel

**Stückl, Pepi i Sojer, Georg** (1993). *Manual completo de montaña* (1a Edició). España: Ediciones Desnivel



## Annex: Instal·lació del projecte

Seràn necessàries algunes eines per poder executar l'aplicació a l'emulador fent servir un servidor de dades local. A continuació es resumeixen els passos necessaris per poder muntar tot l'entorn d'execució.

### Instal·lació de la JVM

---

El primer que s'ha d'instal·lar es una màquina virtual de Java, tant el gestor de base de dades eXist com l'eclipse estan escrits en java.

#### *Instal·lació per GNU/Linux*

La manera més fàcil de fer la instal·lació a GNU/Linux és a través dels seus repositoris. Depenent de la màquina que es vulgui instal·lar s'executarà una línia o l'altra en un terminal de text.

Per l'openjdk (que pot donar problemes amb els diagrames del Protégé):

```
sudo apt-get install openjdk-6-jdk
```

Per la JVM de Sun Microsystems (ara Oracle):

```
sudo apt-get install sun-java6-jdk
```

#### *Instal·lació per Windows o altres*

Descarregar la JVM de la web:

```
http://www.oracle.com/technetwork/java/javase/downloads/jdk-6u25-download-346242.html
```

Primer s'ha d'acceptar la llicència i després ja es pot descarregar la JVM corresponent amb la plataforma on s'hagi d'instal·lar.

Un cop descarregat l'instal·lador s'executa i es segueix l'assistent.

### Instal·lació del gestor de base de dades eXist

---

El gestor de base de dades eXist està escrit en Java, així que hi ha un únic instal·lable per a tots els sistemes operatius, excepte per a la versió Vista del sistema Windows.

## **Instal·lació per Windows Vista**

Descarregar l'instal·lador:

```
http://sourceforge.net/projects/exist/files/Stable/1.4/eXist-setup-1.4.0-rev10433.exe/download
```

i seguir l'assistent d'instal·lació<sup>5</sup>.

## **Instal·lació per GNU/Linux o altes versions de Windows**

Descarregar l'instal·lador:

```
http://sourceforge.net/projects/exist/files/Stable/1.4/eXist-setup-1.4.0-rev10433.jar/download
```

i des d'un terminal de text navegar fins al directori on estigui l'arxiu descarregat i executar la línia:

```
java -jar eXist-setup-1.4.0-rev10440.jar
```

on “eXist-setup-1.4.0-rev10440.jar” es l'arxiu descarregat.

Durant el procés d'instal·lació es demanarà, a part del directori d'instal·lació, la contrasenya per a l'administrador.

## **Instal·lació de l'eclipse**

---

L'eclipse es pot instal·lar igual en qualsevol sistema, encara que els usuaris de GNU/Linux el poden trobar també als repositoris.

## **Instal·lació per GNU/Linux**

A un terminal de text executar la línia:

```
sudo apt-get install eclipse
```

## **Instal·lació per Qualsevol plataforma**

Accedir a la web:

```
http://www.eclipse.org/downloads/
```

i descarregar l'Eclipse IDE for Java Developers, el primer de la llista.

Un cop descarregat, descomprimir l'arxiu i llestos. Dins de la carpeta eclipse hi ha un executable “eclipse” que s'encarrega d'iniciar el programa.

---

<sup>5</sup> L'instal·lació per aquesta plataforma no s'ha pogut provar per no disposar d'aquest sistema operatiu.

## Instal·lació de l'SDK d'Android

L'SDK d'Android es un conjunt d'eines per a desenvolupadors, entre les que cal destacar l'emulador d'Android i el DDMS, una eina per poder interactuar amb el dispositiu/emulador quan s'està provant l'aplicació.

Per fer la instal·lació només s'ha de descarregar l'arxiu comprimit corresponent a la plataforma on s'hagi d'instal·lar i descomprimir-lo.

<http://developer.android.com/sdk/index.html>

En el cas de windows, tot i que existeix l'opció de l'arxiu comprimit, es pot descarregar l'instal·lador (opció recomanada) i executar-lo.

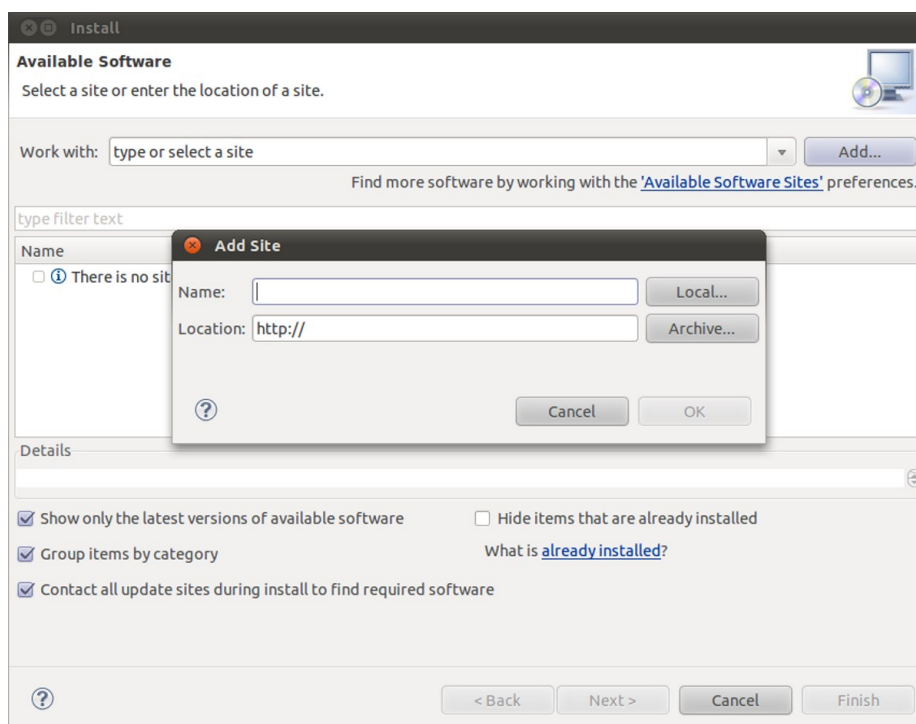
## Instal·lació de l'ADT Plugin for Eclipse

Aquesta eina permet treballar d'una manera més automatitzada des de l'Eclipse, a l'hora de desenvolupar aplicacions Android, que si es fes servir directament l'SDK.

### Instal·lació

Per fer la instal·lació s'han de seguir els següents passos:

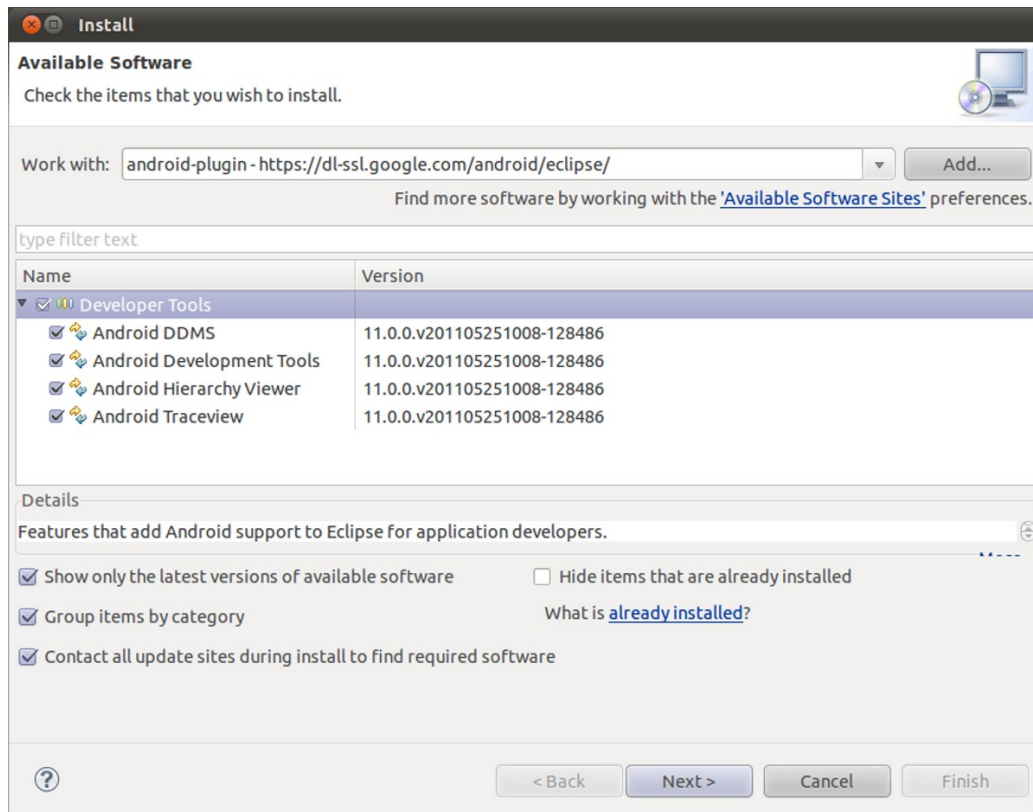
1. Obrir l'Eclipse.
2. Obrir el menú Help/Install New Software...
3. A la finestra que s'ha obert, prémer el botó Add (dalt a la dreta).



- Al diàleg que apareix afegir la línia:

<https://dl-ssl.google.com/android/eclipse/>

- Seleccionar tots els elements per instal·lar

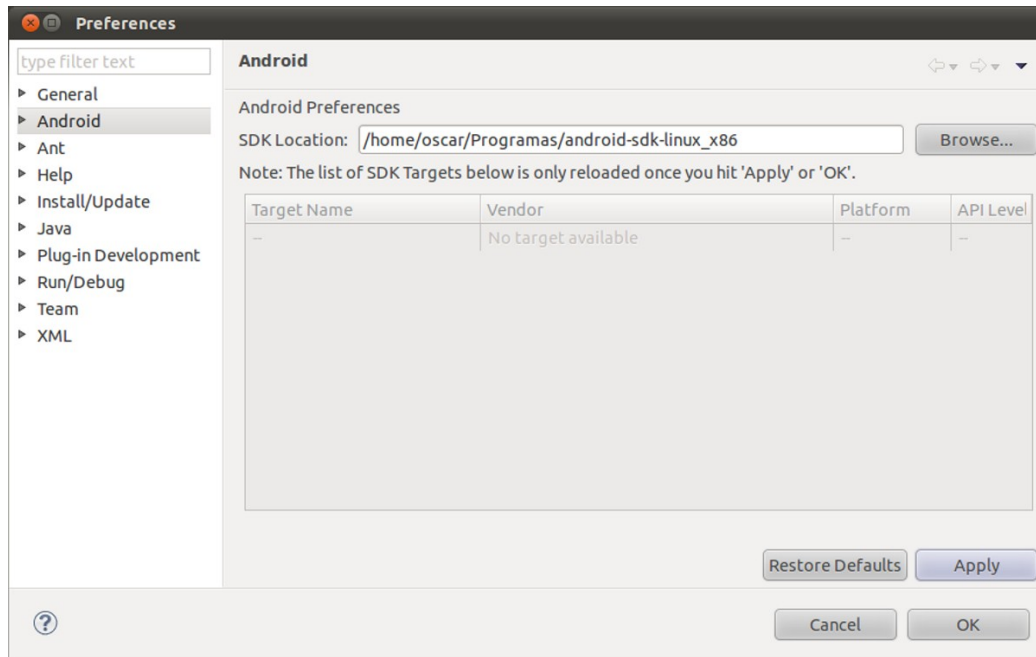


- Seguir l'assistent d'instal·lació (en algun moment s'ha d'acceptar la llicència).
- Reiniciar l'Eclipse.

## Configuració

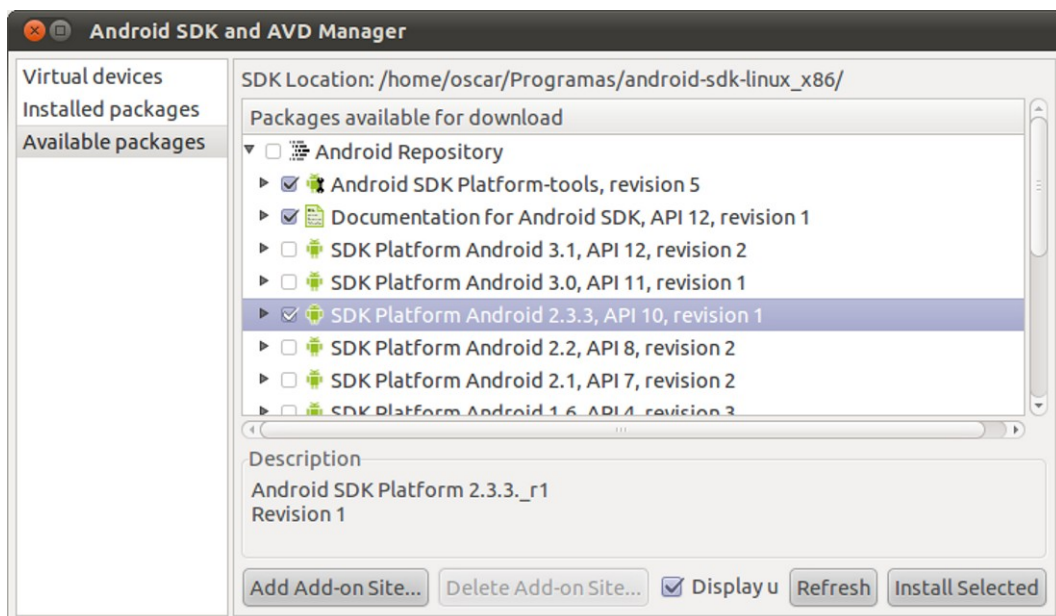
Després de reiniciar l'Eclipse, se li ha de dir on està l'SDK d'Android. Per fer-ho s'ha de:

- Accedir al menú: Window/Preferences
- A la llista esquerra seleccionar l'opció Android.
- Prémer el botó Browse i cercar el directori on s'ha descomprimit l'SDK.



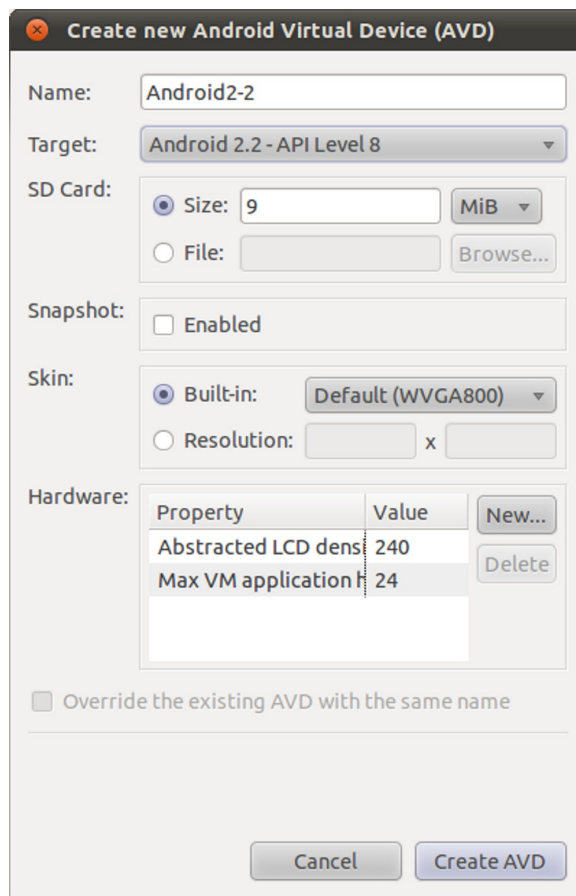
El següent pas és afegir com a mínim un SDK Platform. Aquest són els passos:

1. Obrir el menú: Android SDK and AVD Manager.
2. A l'esquerra, seleccionar: Available Packages
3. Obrir la llista: Android Repository
4. Triar com a mínim un SDK Platform. La última versió del sistema operatiu que existeix en els dispositius reals es la 2.3. encara que per provar l'aplicació amb la 2.2 (API level 8) es suficient.
5. Premer el botó: Install Selected. Durant el procés d'instal·lació s'haurà d'acceptar la llicència.



I per acabar s'ha de crear un Virtual device:

1. Obrir el menú: Android SDK and AVD Manager.
2. A l'esquerra, seleccionar: Virtual devices
3. Prémer el botó New...
4. Configurar el nou dispositiu virtual. Bàsicament s'ha de donar un nom i triar un target d'al menys API Level 8.



5. Per finalitzar, prémer el botó Create AVD.

## Desplegament del projecte

Un cop tot instal·lat, ja es pot fer el desplegament del projecte, que constarà dels passos: (1) pujar l'ontologia al gestor de base de dades, (2) afegir les consultes XQuery a la zona d'aplicacions de l'eXist i (3) importar el projecte Android des de l'Eclipse.

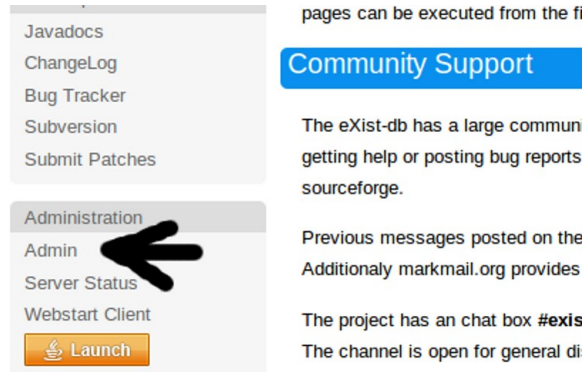
## Ontologia

1. Arrancar el servidor eXist.

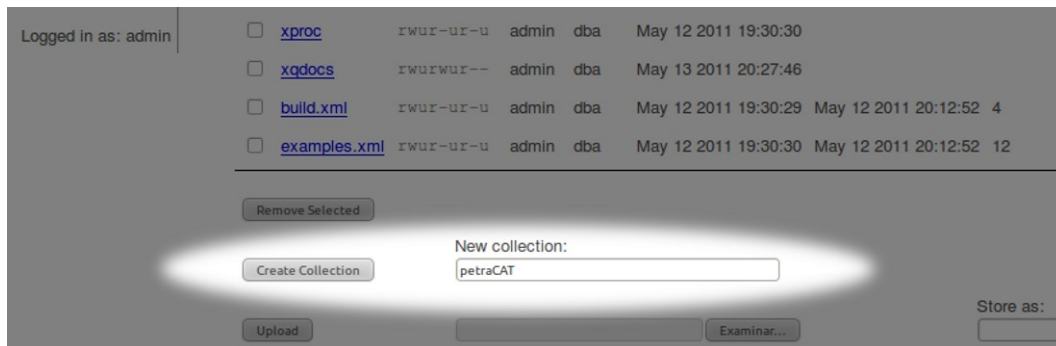
2. Obrir el navegador i, a la barra d'adreces, escriure:

<http://localhost:8080/exist>

3. Accedir a l'àrea d'administració:



4. Introduir l'usuari: admin, i la contrasenya triada en el moment de la instal·lació.
5. Al menú esquerra, fer clic a l'opció: Browse Collection
6. Crear la col·lecció “petraCAT”



7. Accedir al contingut de la col·lecció creada (fent clic a sobre).
8. Pujar l'ontologia, l'arxiu petraCATontologi.owl.

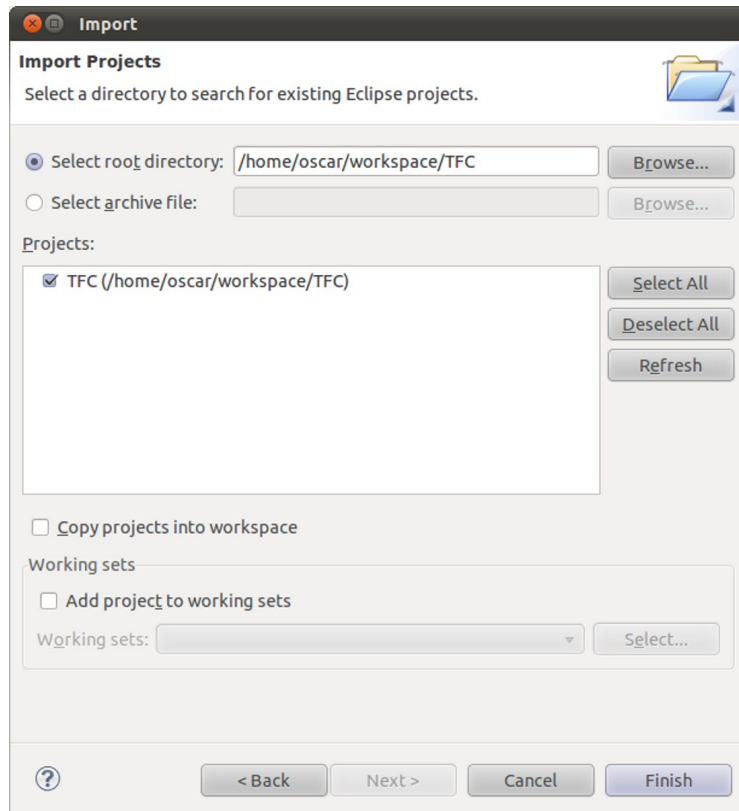
## Consultes XQuery

Copiar totes les consultes de la carpeta xquery a la carpeta webapp/xquery que hi ha dins del directori d'instal·lació de l'eXist.

## Eclipse

1. Descomprimir l'arxiu TFC.zip dins de la carpeta workspace de l'Eclipse.
2. Obrir l'Eclipse.
3. Seleccionar l'opció del menú: File/Import...

4. Dins de la categoria General, triar: Existing Projects into Workspace
5. Prémer el botó: Next>
6. Com a root directory, seleccionar la carpeta TFC.
7. Marcar el projecte TFC de la llista de projectes.



8. Clic al botó Finish.

## Execució del programa

---

Abans d'executar el programa, s'ha de donar la ip de l'ordinador on s'estigui executant el servidor eXist. Per fer-ho, dins de l'Eclipse, editar la classe RemoteData.java (osc.petra.logic) i modificar la constant ipServer.

Ara ja es pot arrancar el programa. Per fer-ho, obrir el menú contextual del projecte TFC i seleccionar l'opció: Run as/Android application

## Enviar coordenades GPS a l'emulador

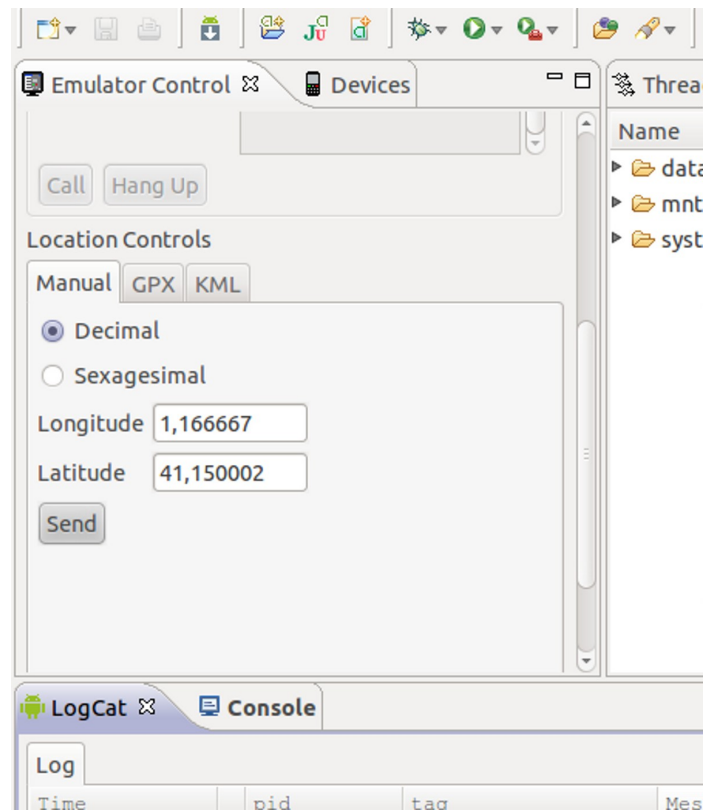
---

Es poden enviar coordenades GPS a l'emulador des de el DDMS. Aquests són els passos a seguir:

1. Tenir l'emulador funcionant.
2. Obrir el menú: Window/Open Perspective/Other...



3. A la llista de perspectives triar: DDMS
4. Buscar dins del tab Emulator Control, a la part esquerra de la pantalla, la secció Location Controls.



5. Escriure les coordenades que es vulguin enviar a l'emulador i fer clic a sobre del botó Send.

# Pòster del diagrama de classes

