ONTOLOGIA DE LINKEDIN: EL PRIMER PAS PER INFERIR INFORMACIÓ A PARTIR DE LES OFERTES DE FEINA

Eva Maria Martín Gómez
Enginyeria Tècnica d'Informàtica de Gestió
Treball Fi de Carrera
Curs 2014-2015 – Primer Semestre

Consultor: **Joan Anton Perez Braña**

FITXA DEL TREBALL FINAL

Títol del treball:	Ontologia de Linkedin: El primer pas per inferir informació a partir de les ofertes de treball
Nom de l'autor:	Eva Maria Martín Gómez
Nom del consultor:	Joan Anton Perez Braña
Data de lliurament (mm/aaaa):	01/2015
Àrea del Treball Final:	Web Semàntica
Titulació:	Enginyeria Tècnica d'Informàtica de Gestió

Resum del Treball (màxim 250 paraules):

Amb aquest treball de final de carrera de la titulació d'enginyeria tècnica en informàtica de gestió es pretén fer una primera aproximació al món de l'anàlisi semàntic de webs.

Consisteix, per una banda, en la creació d'una ontologia per emmagatzemar informació provinent de la web de LinkedIn, concretament, companyies i ofertes de feina publicades, de manera que després pugui ser analitzada i permeti filtrar les dades de manera pràctica evitant l'excés d'informació no útil. Per altra banda, el treball inclou el desenvolupament d'una aplicació per l'obtenció de l' informació de la web de LinkedIn de manera automàtica, i un mètode per l' importació a l'ontologia creada.

Abstract (in English, 250 words or less):

This final work of engineering degree in computer management aims to make a first approach to the world of the semantic analysis for web pages.

It consists, on one hand, the creation of an ontology to store information from the LinkedIn website, specifically, published companies and job offers, so that they can be analyzed and used to filter data in a practical way to avoid the overload of information that is not helpful. Furthermore, the work includes the development of an application for obtaining the information from the LinkedIn website automatically, and a method to import it into the created ontology.

Paraules clau (entre 4 i 8):

1. Català: Web Semàntica, Ontologia, LinkedIn, API, XML, Oferta de feina, Companyia
2. Anglès: Semantic Web, Ontologia, LinkedIn, API, XML, Job, Company

Índex

1. Introducció

Aquest document recull una memòria d'un Treball de Fi de Carrera dels estudis d'Enginyeria Tècnica d'Informàtica de Gestió. En aquest, es defineixen els objectius a assolir amb l'elaboració d'aquest TFC, tant des del punt de vista d'elaboració del mateix, com de la temàtica a tractar.

Un cop establerts els objectius i definit l'abast del projecte, es defineixen les tasques a portar a terme amb les seves precedències entre elles i, en funció del calendari, es fa una planificació del temps disponible per tal d'assolir cadascuna de les fites i dates d'entregues marcades al pla d'estudi de l'assignatura TFC.

Hi ha uns apartats dedicats a la descripció d'aspectes de disseny i desenvolupament de projecte, i producte final obtingut.

L'apartat conclusions inclou una descripció dels coneixements adquirits amb la realització del projecte, anàlisi crítica del seguiment de la planificació i metodologia al llarg del projecte, i una reflexió sobre l'assoliment dels objectius plantejats inicialment.

Per últim, i no per això menys importants, el Glossari i la Bibliografia.

1.1. Context i justificació del Treball

La web actual està orientada, bàsicament, a representar i intercanviar documents, sense cap estructuració de la informació, i això comporta problemes de sobrecàrrega d'informació i heterogeneïtat de fonts d'informació, de manera que la recerca resulta tediosa i poc fiable .

La Web Semàntica pretén ser una extensió de la web actual, però dotant de major significat al seu contingut. Les tecnologies de la web semàntica tenen com a finalitat desenvolupar una web més cohesionada, on sigui més fàcil localitzar, compartir i integrar informació i serveis per tal de maximitzar l'aprofitament dels recursos disponibles a la web.

El precursor de la idea va ser el propi inventor de la www (World Wide Web), Tim Berners-Lee[1], director de la *World Wide Web Consortium* (W3C), una organització dedicada al desenvolupament i manteniment de tecnologies (especificacions, guies, aplicacions i eines) per dotar a la web de més potencial.

Berners-Lee proposa un entorn on a través d'un processament de la informació degudament estructurada, les aplicacions, i per tant les màquines, siguin capaces de realitzar accions d'interpretació de significats de les paraules, de detecció de relacions de sinonímia entre paraules dins un context temàtic concret, de recuperació d'informació relacionada associada a termes no indicats explícitament en una recerca, d'aplicació de certa lògica als resultats a mostrar i de classificació de la informació segons la seva veracitat. És a dir, una web capaç de comprendre i assimilar el contingut de la informació i proporcionar automàticament resultats més precisos, tot afavorint l'experiència de navegació del usuari.

Un dels components de la Web Semàntica és el model RDF[2] (*Resource Description Framework),* un sistema d'etiquetes i regles per al marcatge i anotació de la informació que impliquen la necessària estructuració dels continguts. Aquestes definicions i marcatges que segueixen unes normes establertes, donen lloc a ontologies.

Una ontologia es una representació de coneixement o informació compartida per una comunitat sobre un domini, i expressada en un esquema que permet la interpretació per un sistema informàtic.

Tota ontologia es composa del següents elements:

- **Conceptes o entitats**: són les categories o classes més rellevants al domini.
- **Relacions**: són les connexions semàntiques entres els conceptes. Poden ser relacions d'herència, d'instanciació, de pertinença, etc.
- **Atributs**: són les propietats i valors que s'assignen als conceptes o a les instàncies.
- **Instàncies**: són objectes concrets que existeixen dins del domini.

Segons el nivell de definició que fan dels conceptes les ontologies poden ser de diferents tipus:

- **Alt nivell**: descriuen conceptes generals.
- **De domini**: descriuen vocabulari relacionat amb un domini genèric, especialitzat i/o instanciant conceptes introduïts en ontologies d'alt nivell.
- **De tasques o activitats**: descriuen vocabulari especialitzat d'una tasca o activitat genèrica per mitjà de l'especialització de les ontologies d'alt nivell.
- **D'aplicació**: descriuen conceptes que depenen de les ontologies de domini o de tasques; i sovint són especialitzacions de totes dues. Depenen d'un àmbit concret i dels requisits d'una aplicació en particular.

Avui en dia les xarxes socials estan tenint un gran impacte en tots els aspectes que envolten el nostre dia a dia: amistats, feina, formació, etc. Les xarxes socials cada cop donen més cobertura a aspectes professionals: líders d'opinions, networking, classificació d'empreses per sector, ofertes de feina, etc. Aquestes xarxes socials tenen mecanismes per obtenir la seva informació, cosa que facilita el seu anàlisis per tercers amb l'objectiu de construir sistemes enfocats a realitzar un filtratge de dades més acurat, evitant els problemes d'infoxicació. Una eina d'aquest tipus enfocada a analitzar la informació d'ofertes de feina permetria analitzar les ofertes de feina que s'ofereixen, quines són les empreses que les ofereixen i el potencial interès de cada oferta per un usuari donat.

Dissenyar i crear una eina com la descrita en el paràgraf anterior es el resultat que es pretén obtenir amb la realització d'aquest treball, tot fent servir una ontologia per emmagatzemar la informació més rellevant de les ofertes de feina difoses a LinkedIn i de les companyies que les ofereixen.

En el cas particular que ens ocupa, la ontologia és de tipus domini. Més endavant es detallarà el domini i l'abast d'aquesta.

1.2. Objectius del Treball

L'elaboració d'aquest Treball de Fi de Carrera (TFC) té dos tipus de objectius. D'una banda, els propis de l'assignatura, l'elaboració d'un pla de treball; de l'altra, crear una ontologia amb la informació més rellevant sobre les ofertes de feina i companyies de la DB de LinkedIn; i crear un programa que permeti poblar aquesta ontologia a partir de la informació extreta directament de la web de LinkedIn. Finalment, s'aplicaran una sèrie de regles i es faran consultes per analitzar la informació obtinguda.

1.2.1. Objectius de l'assignatura

L'objectiu d'aquest assignatura és la realització d'un treball de síntesi dels coneixements adquirits en diferents assignatures. El TFC té com a objectiu principal mostrar l'assoliment d'aprenentatge que s'ha dut a terme al llarg dels estudis de l'Enginyeria Tècnica en Informàtica de Gestió.

Aquests objectius es concreten en:
- Analitzar un problema complex de tipus pràctic transformant-lo en un projecte informàtic.
- Planificar i estructurar el desenvolupament del projecte mitjançant l'elaboració d'un pla de treball aplicant una metodologia adient.
- Treballar a fons els aspectes formals del desenvolupament de projectes.
- Sintetitzar una solució viable i realista al problema proposat.
- Elaborar una memòria del projecte segons una estructura prefixada.
- Elaborar una presentació del desenvolupament i resultats finals del projecte.

1.2.2. Objectius del TFC

L'objectiu d'aquest treball es crear un sistema que permeti emmagatzemar informació d'ofertes de feina i les empreses que les ofereixen en una ontologia i explotar aquesta informació. Per tant, el treball tindrà els següents objectius:

1. Crear una ontologia per emmagatzemar ofertes de feina (també es podrà reutilitzar ontologies existents si es troba adient)
2. Crear un programa que permeti poblar l'ontologia a partir de les ofertes de feina (*Jobs*) i les empreses (*Companies*) de LinkedIn.

1.3. Enfocament i mètode seguit

Per tal de poder assolir els objectius del TFC es planteja l'estratègia següent:
- Estudiar OWL (*Web Ontology Language*) i analitzar els tres llenguatges que ofereix (Lite, DL, Full) per decidir quin serà més apropiat per la creació de l'ontologia d'ofertes de feina.
- Aprendre a treballar amb el programari PROTEGÉ que es el que trobo més adient per el desenvolupament del treball, degut a que semblar ser el més utilitzat i millor valorat per desenvolupadors, i per tant amb més documentació.
- Estudiar les APIs de LinkedIn, analitzar l'estructura dels registres Job i Company de LinkedIn per avaluar quina pot ser la informació més rellevant a tenir en compte per

incloure-la a l'ontologia; també considerar els possibles valors i restriccions de cada una de les dades.
- Dissenyar i crear la ontologia pròpiament.
- Implementar una aplicació per a la importació d'instàncies des de LinkedIn a la ontologia. Com a llenguatge de programació es farà servir C#

1.4. Planificació del projecte

1.4.1. Temporització i fites

En aquest punt es fa un estudi de cadascuna de les tasques que s'han de portar a terme, el temps disponible i, en funció d'aquestes dades i de les limitacions temporals que presenta aquest treball (dates de lliurament de les PAC's), s'estableix la planificació i es marquen les fites que s'hauran de complir.

Les tasques es defineixen en funció dels diferents lliuraments que marca el pla d'estudis de l'assignatura. Això és així pel fet que sigui el "client" el que determina unes dates límit, que requereixen un nivell de desenvolupament concret. Dins de cada lliurament, s'avalua la càrrega de feina que serà possible desenvolupar, amb el que queden definits els subapartats.

1.4.2. Calendari

Tenint en compte la disponibilitat horària, en termes generals, 2 hores els dimarts, dimecres i dijous, i 3 hores els dissabtes, diumenges i festius. Amb això podem obtenir el següent calendari (en fons blau, dedicació de dos hores; en fons verd, dedicació de tres hores; en fons vermell i taronja, les fites):

Imatge 1: Calendari planificació projecte

Això vol dir que disposem de:

Setembre	21 hores
Octubre	52 hores
Novembre	54 hores
Desembre	56 hores
Gener	17 hores
TOTAL	**200 hores**

1.4.3. Planificació

Partint d'aquestes hores de què es disposa, es pot preveure la següent planificació:

Tasca	Dates	Hores
1. Definició del projecte	17/09 -> 26/09	13
1.1. Descarregar documentació: descarregar de les bústies de l'assignatura l'enunciat i tota la documentació facilitada pel consultor.	18-sep	
1.2. Llegir documentació: lectura detallada del pla d'estudis i de l'enunciat del treball, així com de tota la informació addicional aportada pel consultor.	18/09 --> 24/09	
1.3. Lectura del mòdul 1.	25/09 --> 26/09	
2. PAC 1: elaboració del pla de treball.	27/09 --> 30/09	8
2.1. Recerca d'informació: cerca a biblioteques, catàlegs digitals, cercadors d'Internet, etc. de tots els temes a tractar en el treball. Intentar tenir tota la documentació necessària abans de començar el desenvolupament del treball pròpiament dit.	27-sep	
2.2. Recollir bibliografia: recollir a la biblioteca la bibliografia recomanada al pla d'estudis i a l'enunciat del TFC, així com la bibliografia trobada a la tasca anterior. La bibliografia recomanada al pla d'estudis i al propi enunciat del treball és considerat material necessari pel bon desenvolupament d'aquest treball. Recollir també altres llibres sobre els temes a tractar que es puguin considerar d'interès.	27-sep	
2.3. Pla de treball: redacció dels diferents punts que formen el pla de treball. Això és, objectius, guió, apartats que contindrà el treball, número de planes per apartat; incidències, riscos i els corresponents plans de contingència; fites (incloent els lliuraments de les Pac's) i avaluació del material necessari. Deixar la temporització per quan estiguin definits tots els altres punts.	28-sep	
2.4. Temporització: un cop avaluat tot l'abast del treball, analitzar el temps necessari per realitzar cadascuna de les tasques, el temps disponible per portar-les a terme i fer la temporització, amb el corresponent diagrama de Gannt.	29/09 --> 30/09	
3. Lliurament de la PAC 1	30-sep	
4. Instal·lació: instal·lació del programari necessari pel bon desenvolupament d'aquest treball. Donat que al punt de treball ja es disposa de Microsoft Word, Microsoft Power Point i Gantt Project, només restarà instal·lar el programa PROTÉGÉ. Instal·lar el software necessari també a l'ordinador de la feina, com a mesura de precaució.	01-oct	2
5. Esborrany PAC 2: elaboració de l'esborrany de la PAC 2.	02/10 --> 29/10	48
5.1. Lectura dels mòduls 2 i 3	02/10 --> 05/10	
5.2. Aprofundir en l'estudi de la eina Protegé i el llenguatge OWL. Estudi de la documentació dels links proposats al document "Orientacions_TFC_XML-Web_semàntica.pdf" (facilitada pel consultor) sobre les APIs de Linkedin i l'estructura de les ofertes de feina i les companyies.	07/10 --> 14/10	
5.3. Creació del conjunt de metadades.	14/10 --> 18/10	
5.4. Disseny i desenvolupament, amb Protegé, de l'ontologia que permeti definir un conjunt d'ofertes de feina, detalls sobre les ofertes i les companyies que les ofereixen.	18/10 --> 28/10	
5.5. Lliurament esborrany PAC 2	29-oct	
6. Correcció PAC 2 en funció de les anotacions del consultor i completar els punts tractats a la PAC 2	30/10 --> 04/11	10
7. Lliurament PAC 2	04-nov	
8. Esborrany PAC 3	05/11 --> 01/12	46
8.1. Aprofundir en l'estudi de les API's de Linkedin.	05/11 --> 09/11	
8.2. Anàlisi i disseny	10/11 --> 30/11	
8.3. Lliurament esborrany PAC 3	01-dic	
9. Correcció PAC 3 en funció de les anotacions del consultor i completar els punts tractats a la PAC 3	02/12 --> 09/12	14
10. Lliurament PAC 3	09-dic	
11. Conclusions: redacció de les conclusions del treball.	10/12 --> 20/12	19
12. Revisió final: lectura acurada del treball, revisant ortografia, redacció i els propis continguts.	21/12--> 24/12	7
13. Síntesi: selecció de les parts del treball que seran d'utilitat per a la realització d'una presentació.	25/12 --> 27/12	9
14. Presentació: desenvolupament d'una presentació de 20 diapositives més veu fetes amb Microsoft Power Point, amb una síntesi del treball.	28/12 --> 09/01	24
15. Lliurament memòria i presentació.	09-ene	

Taula 1: Planificació de tasques

1.4.4. Diagrama de Gantt

El diagrama de Gantt que es representa la planificació establerta és el següent:

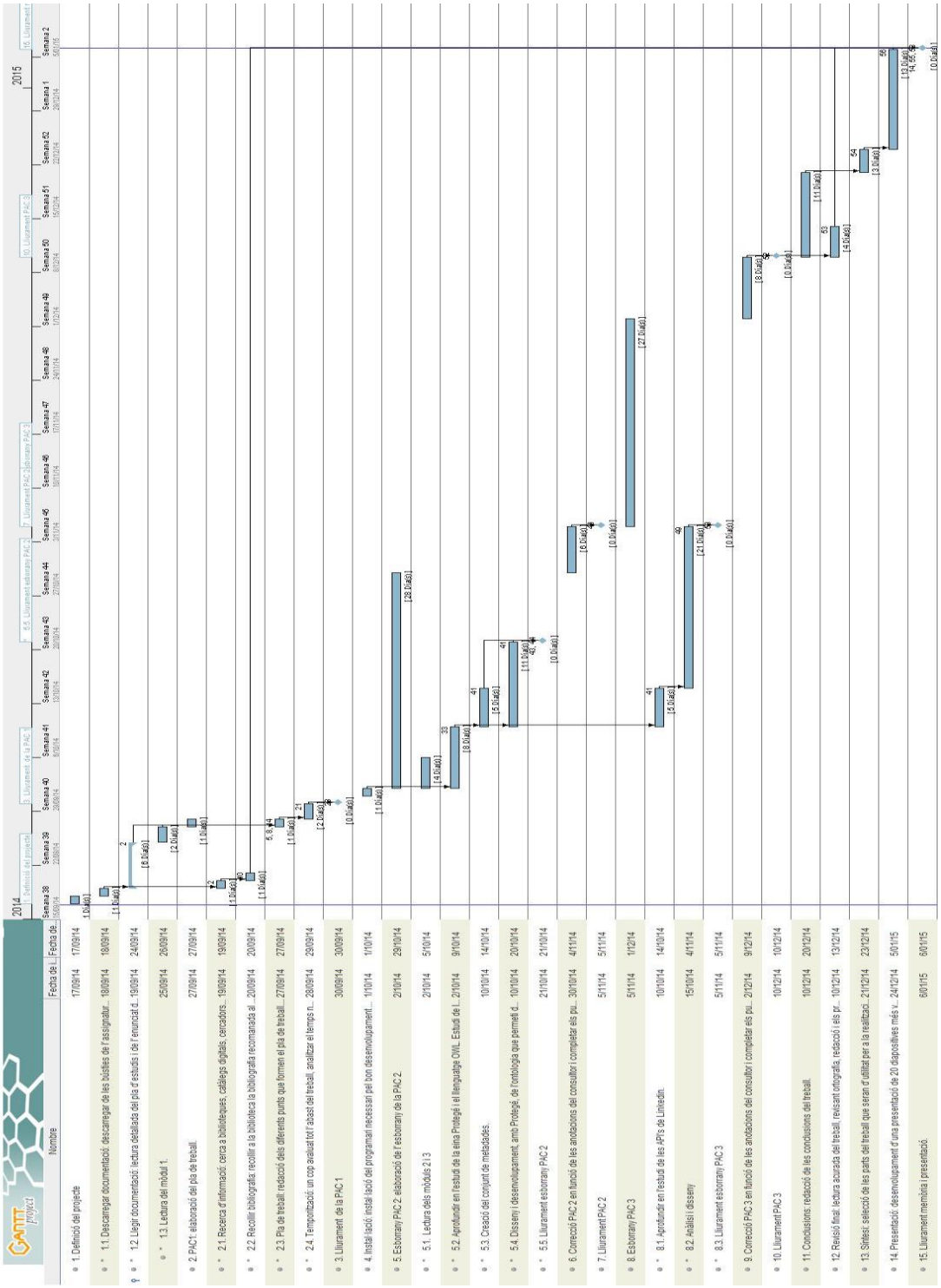


Diagrama 1: Gantt de planificació del projecte

1.5. Breu sumari de productes obtinguts

S'ha obtingut el conjunt de metadades d'una Ontologia en OWL creada amb Protégé v. 3.4.8, per l'emmagatzematge de feines i companyies que consten a la web de LinkedIn, i algunes instàncies per exemplificar la seva funcionalitat.

Per tal de poblar l'Ontologia s'ha creat un petit programa que connecta al domini LinkedIn, recull una mostra d'ofertes de feina i empreses, i genera uns fitxers XML que podran ser importats a l'ontologia per crear instancies automàticament.

També s'ha creat un manual d'usuari que descriu com instal·lar el programa per l'extracció de dades de LinkedIn i la posterior importació d'aquestes a la ontologia.

1.6. Breu descripció dels altres capítols de la memòria

En el capítol de Construcció d'una ontologia s'inclou, en subcapítols, cadascuna de les tasques realitzades per el disseny i la creació de l'Ontologia.

En el capítol Creació d'una aplicació per poblar l'Ontologia es troba la descripció de l'anàlisi, disseny i creació del programa per a la recol·lecció d'informació per generar instàncies a l'ontologia.

2. Construcció de l'ontologia

Per arribar a construir una ontologia es imprescindible un bon disseny. Aquest disseny inclou uns punts bàsics. En aquest apartat s'entrarà en detall en cada un d'aquest punts i es farà l'explicació del disseny pròpiament de l'ontologia.

El nom que he donat a l'Ontologia es *OLinkedIn*.

Primerament s'ha establit el domini i l'abast que es pretenia cobrir i s'ha avaluat si d'altres ontologies ja existents podien servir com a base de la nova ontologia. A continuació he fet un llistat de tots els termes que podien ser d'interès sobre el domini establert.

A partir d'aquí ja he pogut passar a la definició de conceptes (classes) i relacions (propietats i restriccions). Després he creat algunes instàncies per realitzar l'especificació de cada classe a mode d'exemple.

2.1. Tecnologia

Com comentàvem en l'apartat de mètode a seguir, l'editor escollit per a la creació de l'ontologia ha estat Protégé.

2.1.1. Protégé[3]

Protégé és una eina Open Source que permet la creació i edició d'ontologies i bases de dades de coneixement. Ha estat desenvolupada principalment per Stanford Medical Informatics amb la col·laboració de la Universitat de Manchester.

Protégé presenta una interfícies amigable per a l'edició d'ontologies, representa les restriccions amb operadors de lògica matemàtica. Es caracteritza també per els anys que porta de desenvolupament continuo i la gran comunitat d'usuaris i desenvolupadors que hi participen en el projecte. Existeixen molts plugins per afegir que aportar diverses funcionalitats. Per exemple, algunes de les que farem servir:

- **Jambalaya**: per generar diagrames representatius de l'ontologia.
- **XML Tab**: per exportar/importar l'estructura de l'ontologia en format XML, per importar instàncies de classes.
- **Queries:** per realitzar consultes sobre les instàncies.

2.1.2. OWL[4]

OWL (Web ontology Language) és un llenguatge marcat, construït sobre RDF i codificat en XML, per a publicar i compartir dades mitjançant ontologies en la World Wide Web (www).

Hi ha tres nivells de llenguatge OWL segons el grau d'expressivitat i complexitat computacional dels raonaments a realitzar. De menys a més expressiu:

• **OWL Lite**: és el subllenguatge més senzill, permet una àgil migració des de altres
llenguatges ontològics. La seva lògica té restriccions numèriques molt limitades.

• **OWL DL**: és més expressiu que l'anterior però un grau de complexitat alt (considerant e pitjor cas). Es el més apropiat per a aplicacions que requereixen el màxim d'expressivitat però exigeixen certa complexitat computacional.

• **OWL Full**: no aporta més sintaxi que el DL però permet un gran canvi conceptual.
Full permet que diferents objectes tinguin mateix identificador (URI).

El llenguatge que he fet servir per l'ontologia és OWL DL.

2.2. Domini i abast

Per crear l'ontologia que ens ocupa, el domini que es vol cobrir son les ofertes de feina i les companyies de LinkedIn. La finalitat de l'ontologia és emmagatzemar la informació més rellevant de les feines i les companyies, i explotar aquest informació.

L'ontologia permet un fàcil accés a les dades, per tal d'analitzar-les i comprendre-les. Per això, s'han plantejat unes preguntes de verificació per determinar l'abast de la nostra ontologia, al mateix temps que serveixen com a control de qualitat de la mateixa. Les preguntes, "competency questions", que ha de permetre respondre l'ontologia per LinkedIn seran:

1. Quines companyies ofereixen una feina?
2. Quins requisits és demanen per una oferta?
3. Quina és la data més propera en la que s' oferta una feina?
4. En quina localització hi ha més ofertes de feina?
5. Quines són les funcions més demandades?
6. Quin tipus d'indústria és la que ofereix més feines?
7. Quin tipus de contracte és el més oferta?
8. Quina funció està millor retribuïda?
9. Quin és el nº d'empleats de les companyies amb més ofertes de feina?
10. Quin és l'historial d'ofertes d'una companyia en un període concret?

2.3. Reutilització

Abans de començar a desenvolupar una ontologia, es convenient considerar la possibilitat de reutilitzar alguna ontologia ja existent, encara que sigui d'un nivell més alt per instanciar algun dels conceptes que formen part de la nostra ontologia.

Amb alguns dels cercadors d'ontologies com Swoogle, Ontolingua i DAML i consultant algunes llibreries online d'ontologies s'ha analitzat si alguna de les ontologies existents es podria fer servir com a base de partida o com a part de l'ontologia d'aquest projecte, però no s'ha trobat quelcom adient.

2.4. Enumeració de termes

En aquest punt s'enumeren tots els termes relacionats amb el domini feines i companyies de LinkedIn que poden ser d'interès en l'ontologia Aquests termes seran el punt de partida per a la definició de les categories (classes), propietats i enunciats que poden aparèixer en respostes a les preguntes de verificació que comentava en l'apartat de domini i abast.

He analitzat com s'estructura la informació de feines i companyies segons la documentació que ofereix LinkedIn per a desenvolupadors als següents links:

https://developer.linkedin.com/documents/job-lookup-api-and-fields (estructura Job)

Parameter	Definition
id	Default. The job ID.
customer-job-code	Default. Customer entered job code.
active	Default. Indicates whether or not this is an active job posting. Boolean.
posting-date	Date of job posting. Format is YearMonthDay or YearMonth.
expiration-date	The expiration for the job posting. Format is YearMonthDay or YearMonth.
posting-timestamp	Default. The timestamp for the job posting. Time is in milliseconds.
expiration-timestamp	The timestamp for the job posting expiration. Time is in milliseconds.
company:(id)	Unique ID for the hiring company.
company:(name)	Name of the hiring company.
position:(title)	Title of the posted position.
position:(location)	Location for the position being posted. Can contain the country code, postal code, and name.
position:(job-functions)	Function for the position. See Job Functions for the list of functions available.
position:(industries)	Function for the position. See Industry Codes for the list of industries available.
position:(job-type)	Job type for the position being posted. See Job Types for the list of types available.
position:(experience-level)	Experience level for the position being posted. See Experience Levels for the list of experience levels available.
skills-and-experience	Description of the skills and experience needed for the posted position
description-snippet	Default. Short description for the position.
description	The full description for the position.
salary	The salary listed for the posted job.
job-poster:(id)	The ID for the person who posted the position.
job-poster:(first-name)	The first name of the person who posted the position.
job-poster:(last-name)	The last name of the person who posted the position.
job-poster:(headline)	The headline title for the posted position.
referral-bonus	Provides information if there is a referral bonus.
site-job-url	The URL for the posted position.
location-description	The description of the position's location.

Taula 2: Estructura de job a LinkedIn

https://developer.linkedin.com/documents/company-lookup-api-and-fields (estructura Company)

Parameter	Definition
id	Default. The unique internal numeric company identifier.
name	Default. The human readable name of the company.
universal-name	The unique string identifier for a company.
email-domains	Company email domains.
company-type	Type of company. Valid values are: ▪ **C** ("Public Company") ▪ **D** ("Educational") ▪ **E** ("Self Employed") ▪ **G** ("Government Agency") ▪ **N** ("Non Profit") ▪ **O** ("Self Owned") ▪ **P** ("Privately Held") ▪ **S** ("Partnership") Use this field instead of the deprecated *type* field.
ticker	Company ticker identification for the stock exchange. Available only for public companies.
website-url	Company web site address.
industries	A collection containing a code and name pertaining to the company's industry. See Industry Codes for the list of industries available.
status	Company status. Valid values are: ▪ **OPR** ("Operating") ▪ **OPS** ("Operating Subsidiary") ▪ **RRG** ("Reorganizing") ▪ **OOB** ("Out of Business") ▪ **ACQ** ("Acquired")
logo-url	URL for the company logo in JPG format.
square-logo-url	URL for the company logo in a square format.
blog-rss-url	URL for the company blog.
twitter-id	Handle for the company Twitter feed.
locations:(contact-info:(phone1))	Company phone number for the location.
locations:(contact-info:(phone2))	Second company phone number for the location.
locations:(contact-info:(fax))	Company fax number for the location.
description	Company description. Limit of 500 characters.
stock-exchange	Stock exchange the company is in. Available only for public companies. Val ▪ **ASE** (1, "American Stock Exchange") ▪ **NYS** (2, "New York Stock Exchange") ▪ **NMS** (3, "NASDAQ") ▪ **LSE** (4, "London Stock Exchange") ▪ **FRA** (5, "Frankfurt Stock Exchange") ▪ **GER** (6, "XETRA Trading Platform") ▪ **PAR** (7, "Euronext Paris")
founded-year	Year listed for the company's founding.
end-year	Year listed for when the company closed or was acquired by another.
num-followers	The number of followers for the company's profile.
employee-count-range	Number range of employees at the company. Use this field instead of the deprecated *size* field. Valid values are: ▪ A:1 ▪ B: 2-10 ▪ C: 11-50 ▪ D: 51-200 ▪ E: 201-500 ▪ F: 501-1000 ▪ G: 1001-5000 ▪ H: 5001-10,000 ▪ I: 10,000+
specialties	Company specialties. Retrieves information from string input.
locations	Company location.
locations:(description)	Description of company location.
locations:(is-headquarters)	Valid values are **true** or **false**. A value of **true** matches the Company headquarters location.
locations:(is-active)	Valid values are **true** or **false**. A value of **true** matches the active location.
locations:(address)	Address of location.
locations:(address:(street1))	First line of street address of location.
locations:(address:(street2))	Second line of street address of location.
locations:(address:(city))	City for location.
locations:(address:(state))	State for location.
locations:(address:(postal-code))	Postal code for location. Matches companies within a specific postal code. Must be combined with the **country-code** parameter. Not supported for all countries.
locations:(address:(country-code))	Country code for location. Matches companies with a location in a specific country.
locations:(address:(region-code))	Region code for location.
locations:(contact-info)	Company contact information for the location.

Taula 3: Estructura de company a LinkedIn

M'he basat en aquesta estructura per a la implementació de les metadades de l'ontologia, però fent una selecció de la informació, no he incorporat tots els camps.

Termes en l'ontologia per el domini LinkedIn:

- Companyia (Company)
  - Id
  - Name
  - Descripció
  - Tipus de companyia i els diferents tipus que s'especifiquen a l'API de LinkedIn
  - Rang de nº d'empleats
  - Sector (Industry) i els diferents tipus que s'especifiquen a l'API de LinkedIn
  - Localitat
  - Nom
  - Estat
  - Website
  - Borsa de valors de la companyia
- Feina (Job)
  - Id
  - Activa
  - Descripció
  - Data de registre
  - Data caducitat
  - Nivell d'experiència i els diferents nivells que es detallen a l'API de LinkedIn
  - Tipus de Contracte i els diferents tipus que s'especifiquen a l'API de LinkedIn Funció i els diferents tipus que s'especifiquen a l'API de LinkedIn
  - Id de la companyia que oferta la feina
  - Nom de la companyia que oferta la feina
  - Data Registre
  - Salari
  - Títol

L' idioma de l'ontologia és l'anglès per tant d'ara en endavant em referiré als conceptes amb la nomenclatura en anglès que els hi he donat dins de l'ontologia.

La nomenclatura per atributs i conceptes en l'ontologia respecta els noms establerts en LinkedIn, en previsió de facilitar la implementació del programa d'extracció de dades, transformació dels fitxers xml obtinguts i la importació després a l'ontologia.

2.5. Definició de conceptes

Prenent com a base el glossari de termes enumerats en l'apartat anterior, he seleccionat aquells conceptes que descriuen objectes independents per tal de constituir les classes. La resta de termes serveixen per crear propietats.

Com a conceptes generals es troben: **company, job, job-function i industry**.

Son les classes principals que s'han definit a l'ontologia com a disjuntes, donat que cada element que representa cada una de elles només pot pertànyer a aquesta classe i no a una altre.

company, és la classe per representar las companyies de LinkedIn.

job, és la classe per representar les ofertes de feina de LinkedIn

job-function, és la classe que representa i enumera les diferents funcions que consten a LinkedIn.

industry, és la classe que representa i enumera la classificació d'indústries que consten a LinkdIn.

Per les classes *company* i *job*, he creat a més, unes subclasses amb unes propietats i valors concrets respecte la classe principal. Es veurà en detall a l'apartat de definició de propietats i restriccions.

Dins de la classe **company** he creat les subclasses :

- **BigCompany**
- **MediumCompany**
- **SmallCompany**
- **PrivateCompany**
- **PublicCompany**

I dins de la classe **job** he creat les subclasses:

- **Engineer**
- **Commercial**
- **Educator**
- **BusinessManagement**
- **Industrial**
- **ProductManagement**
- **Auditor**
- **JobWithRequiredExperience**

Mitjançant Jambalaya, una de les eines per a representació gràfica que ofereix Protégé, creo el diagrama de classes de l'ontologia que queda de la següent manera:

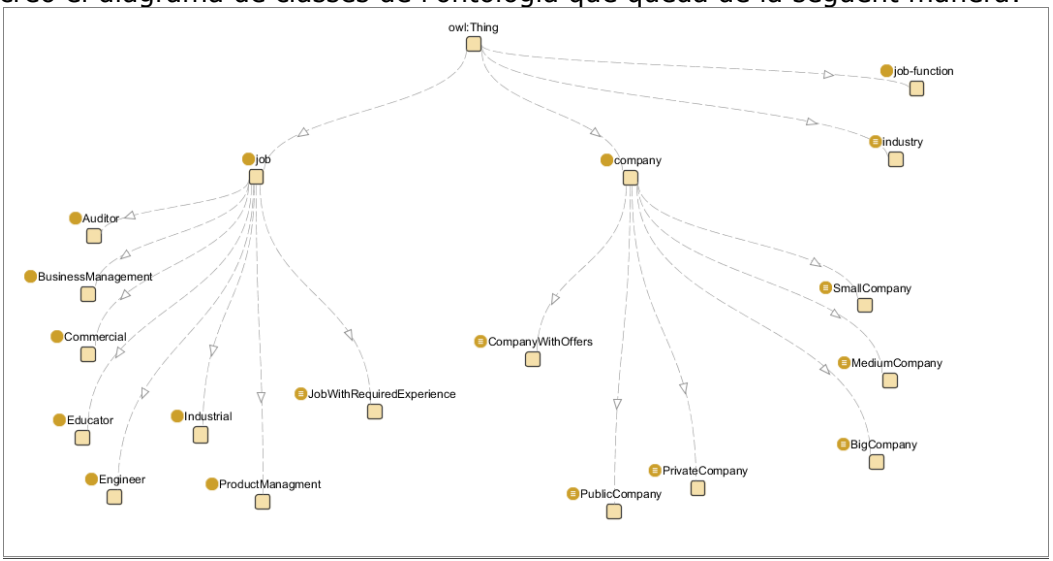


Diagrama 2: Estructura de classes de l'ontologia OlinkedIn

2.6. Definició de propietats i restriccions

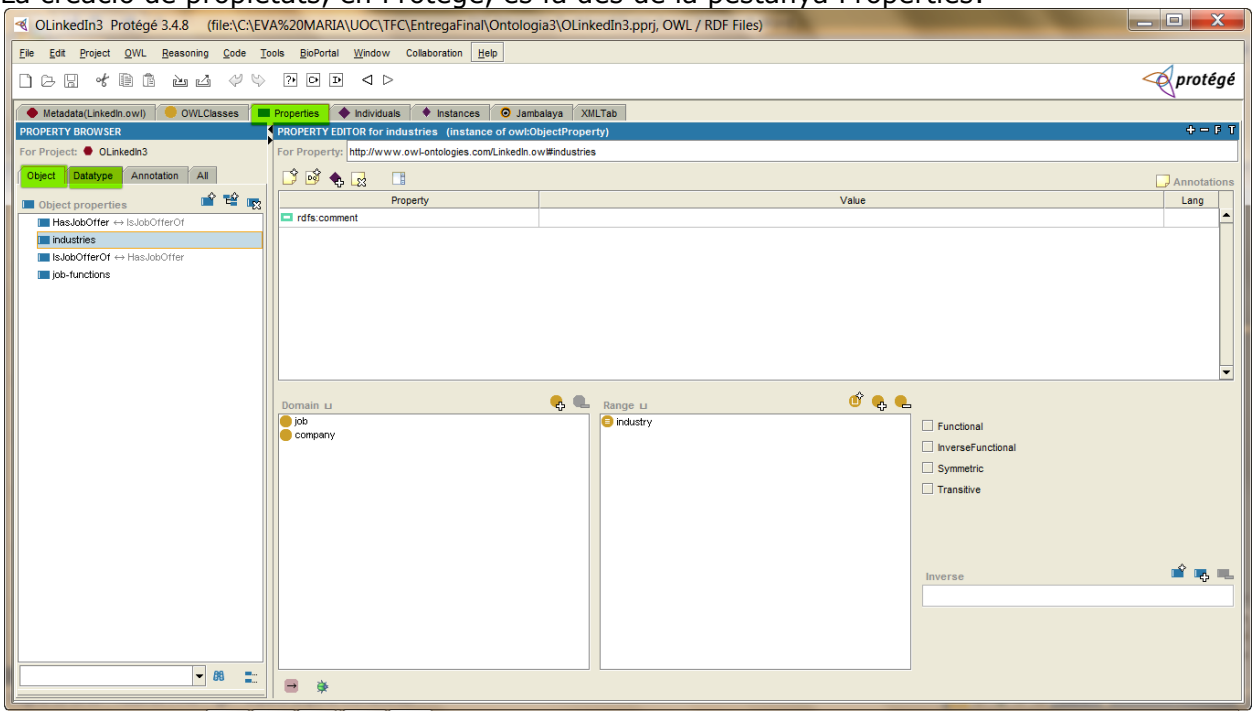
Definides ja les classes es pot passar a la definició dels seus atributs i relacions. Gairebé tota la resta de termes que no han esdevingut una classe, es converteixen ara en propietats. La definició de les propietats o atributs ens permetrà relacionar i descriure conceptes.

En una ontologia, les propietats es caracteritzen per les següents restriccions bàsiques:

- **Tipus de valor:** descriu el tipus de valor que pot adquirir la propietat, com poden ser: decimal, cadena de caràcters, booleà, etc.
- **Domini o Rang:** per determinar o bé el rang de classes permeses o un rang de valors determinats.
- **Cardinalitat:** estableix el nombre de valors que pot tenir la propietat, o el número d'objectes que resulten en la relació d'una classe amb una altre classe. Si només pot agafar una valor llavors tindrà cardinalitat simple, si pot agafar diferents valors llavors la cardinalitat serà múltiple.
- **Quantificació:** permet indicar l'existència d'algun o tots els objectes en la relació d'una classe amb una altre

Per cada classe he definit les propietats de tipus de dades y les propietats d'objecte, algunes funcionals amb la seva inversa corresponen. Les propietats funcionals relacionen com a molt una classe amb una altre. Per cada propietat funcional ha d'existir la seva inversa.

La creació de propietats, en Protégé, es fa des de la pestanya Properties:



Imatge 2: Creació de propietats en Protégé

I dins d'aquest formulari es poden crear les propietats de dades, atributs de cada classe, i les propietats d'objecte que serveixen per relacionar una classe amb d'altres.

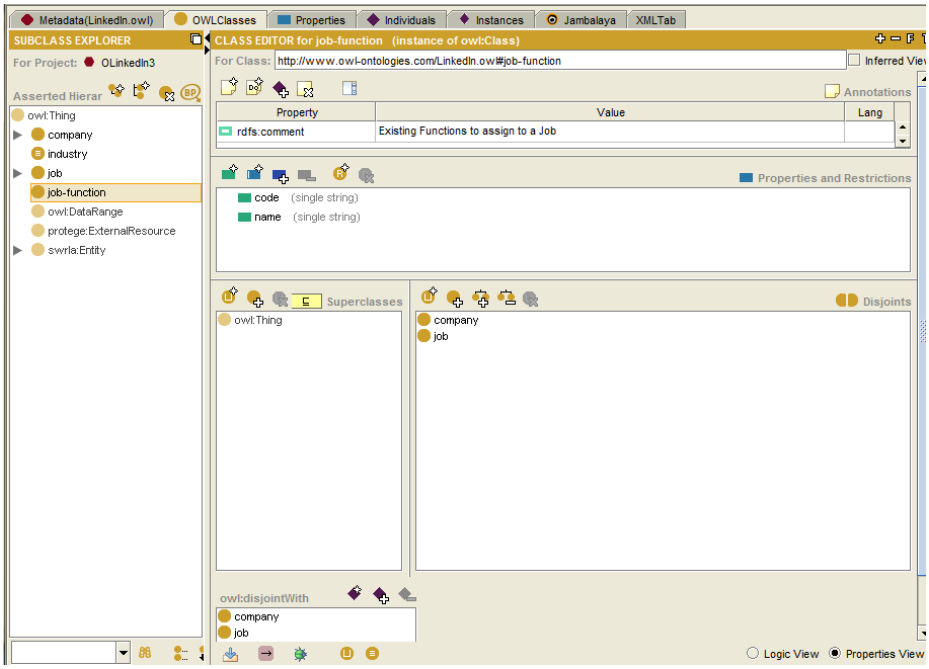
A continuació detallaré com s'han creat les quatre classes principals i les seves subclasses .

Per les classes *industry* i *job-function*, s'han creat com a individus concrets els diferents elements de Industry i de Functions segons informació de la seva estructura obtinguda a LinkedIn.

job-function

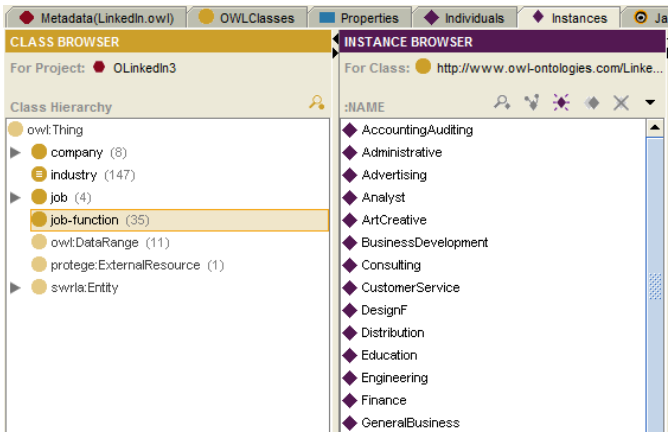
Es la classe els elements de la qual son les diferents funcions de la llista Functions de LinkedIn (**https://developer.linkedin.com/documents/job-functionsSlot**)

Com a atributs té *code* i *name*:



Imatge 3: Definició de la classe job-function

S'enumeren els seus elements creant-los com a individus:

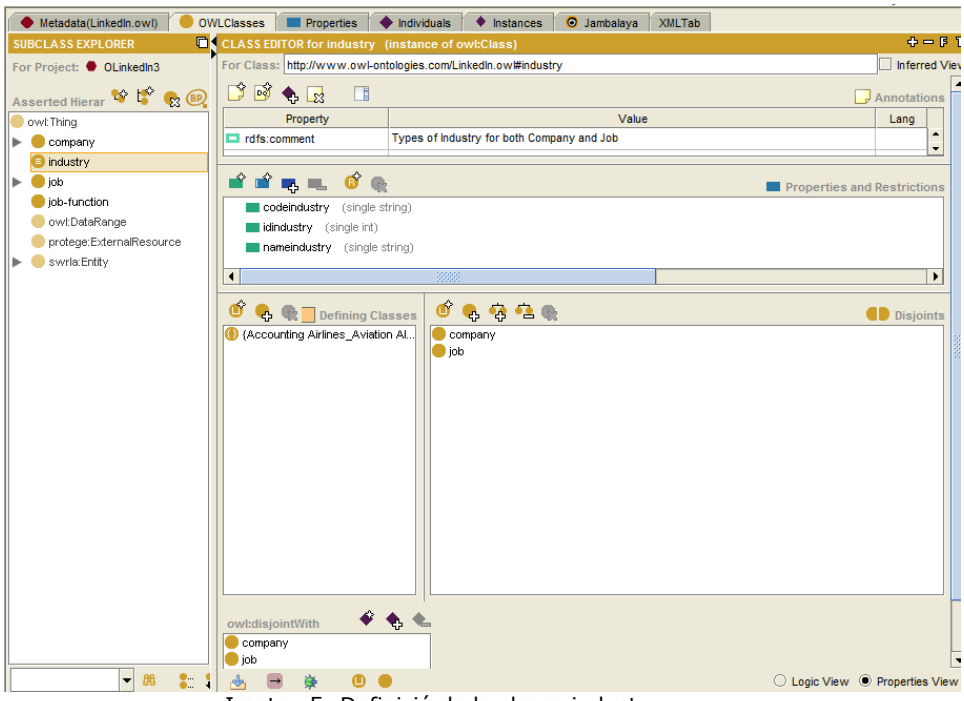


Imatge 4: Elements de la classe job-function

industry

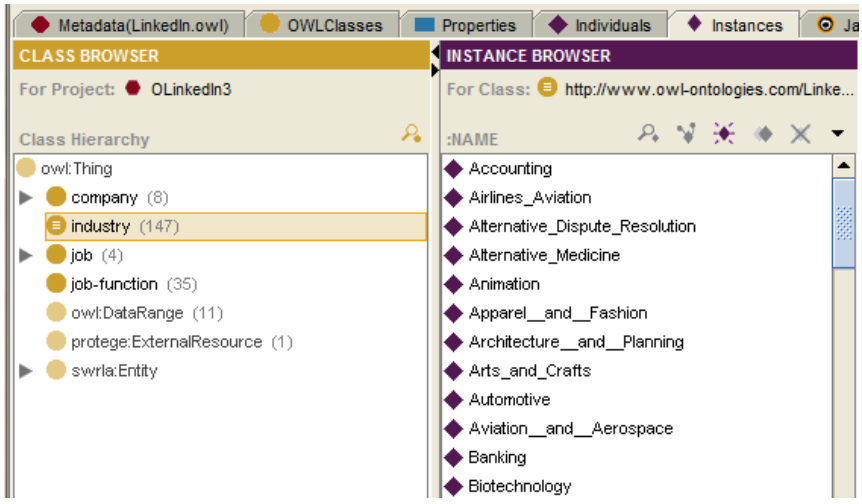
Es la classe els elements de la qual son les diferents funcions de la llista de Industry de LinkedIn (**https://developer.linkedin.com/documents/industry-codes**).

Com a atributs té *codeindustry*, *idindustry* i *nameindustry*.



Imatge 5: Definició de la classe industry

S'enumeren els seus elements creant-los com a individus:

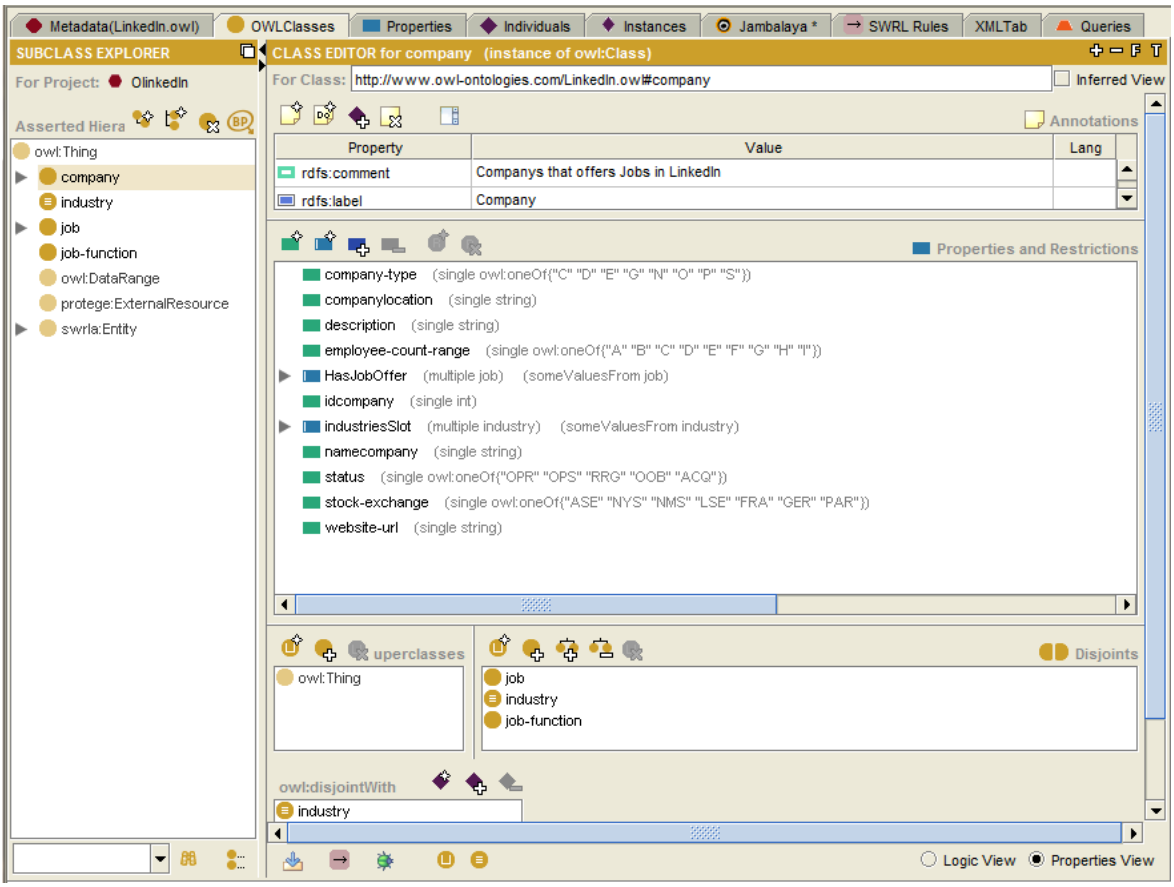


Imatge 6: Elements de la classe industry

company

Consta d'atributs bàsics com id, nom, descripció, etc. i d'altres més específics com tipus de companyia, rang de nombre d'empleats, etc.

En la imatge es poden veure les propietats de la classe *company*, el tipus de valor que admet cada una d'elles i la cardinalitat:



Imatge 7: Definició de la classe company

Per cada un dels atributs *company-type, employee-count-range, status i stock-exchange* he definit el rang de valors que poden tenir en base a l' informació de l'estructura de Company a LinkedIn (veure apartat Enumeració de termes). Els he afegit com a atributs de classe en lloc de crear-les com a classes perquè en si no representen un concepte en concret, sinó que permeten definir una característica d'un concepte. Pràcticament tots els atributs son de cardinalitat simple.

A més dels atributs de dades (DataType) també he creat unes propietat d'objecte (ObjectProperty) per relacionar la classe *company* amb les classes *job* i *industry*. Concretament he creat les propietats:

- **HasJobOffer** per trobar si una companyia té alguna oferta de feina, amb cardinalitat múltiple perquè una companyia pot tenir *n* ofertes de feina.
- **industriesSlot** per relacionar cada companyia amb els diferents valors sobre la classe *industry* que correspongui.

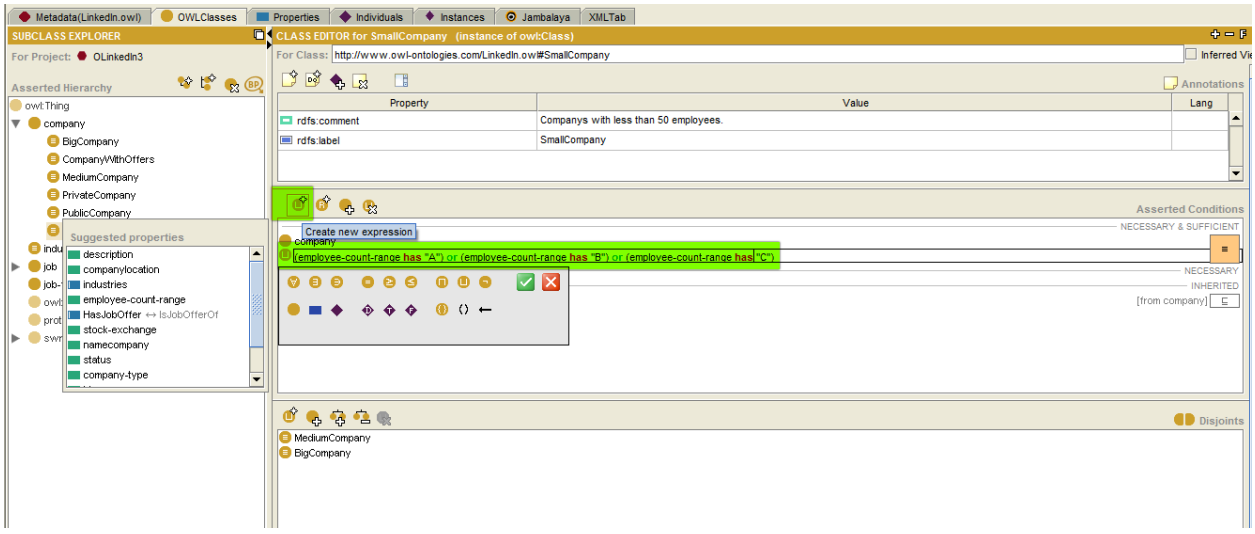
Amb la intenció de permetre una classificació més detallada de cada individu (instància) de companyies, he modelat unes subclasses dins de *company*. La taxonomia queda així:

- **BigCompany** per classificar empreses de més de 1000 empleats.
- **MediumCompany** per classificar empreses entre 50 i 1000 empleats.
- **SmallCompany** per classificar empreses de fins a 50 empleats.
- **PrivateCompany** per classificar empreses de tipus: PrivateHeld, SelfOwned, SelEmployeed i Partnership
- **PublicCompany** per classificar empreses de tipus: Public, Educational, GovernmentAgency i Nonprofit
- **CompanyWithOffers** per classificar empreses que tenen con a mínim una oferta de feina. es basa en una restricció de cardinalitat que sembla que amb OWL en Protégé no acaba de funcionar.

Les tres subclasses *SmallCompany*, *MediumCompany* i *BigCompany* les he definit com a disjuntes entre elles.

Des de l'editor de classes, selecciono el botó *Create new expression,* llavors apareix una finestra amb la llista d'atributs de la classe seleccionada i un altre amb les diferents operacions que es poden realitzar amb aquets atributs. Escric l'asserció per l'atribut *employee-count-range* per determinar quins valors ha de tenir per tal de que una companyia sigui classificada com a Small, que serà, segons la definició que hem apuntat abans i tenint en compte l'estructura a LinkedIn de *company* (veure taula 3 de l'apartat Enumeració de termes), pels valors A, B o C. L'expressió doncs, queda de la següent manera:

(employee-count-range has "A") or (employee-count-range has "B") or (employee-count-range has "C")



Imatge 8: Definició d'una asserció

Seguint el mateix mètode utilitzat per a la definició de la subclasse *SmallCompany*, especifico les assercions per les subclasses *MediumCompany* i *BigCompany*.

Per *MediumCompany* els valors que haurà de tenir l'atribut *employee-count-range*, seràn D, E o F. L'expressió queda de la següent manera:
(employee-count-range has "D") or (employee-count-range has "E") or (employee-count-range has "F")

L'expressió per la definició de la subclasse *BigCompany* queda:
(employee-count-range has "G") or (employee-count-range has "H") or (employee-count-range has "I")

Pel modelatge de les subclasses *PublicCompany* i *PrivateCompany*, s'han definit com disjuntes entre elles i s'ha aplicat restricció sobre l'atribut *company-type*. S'han creat les expressions tenint en compte els possibles valors que pot tenir la propietat *company-type* de l'estructura de company a LinkedIn (veure taula 3 de l'apartat Enumeració de termes) i la definició que s'ha establert per aquestes subclasses dins l'ontologia.

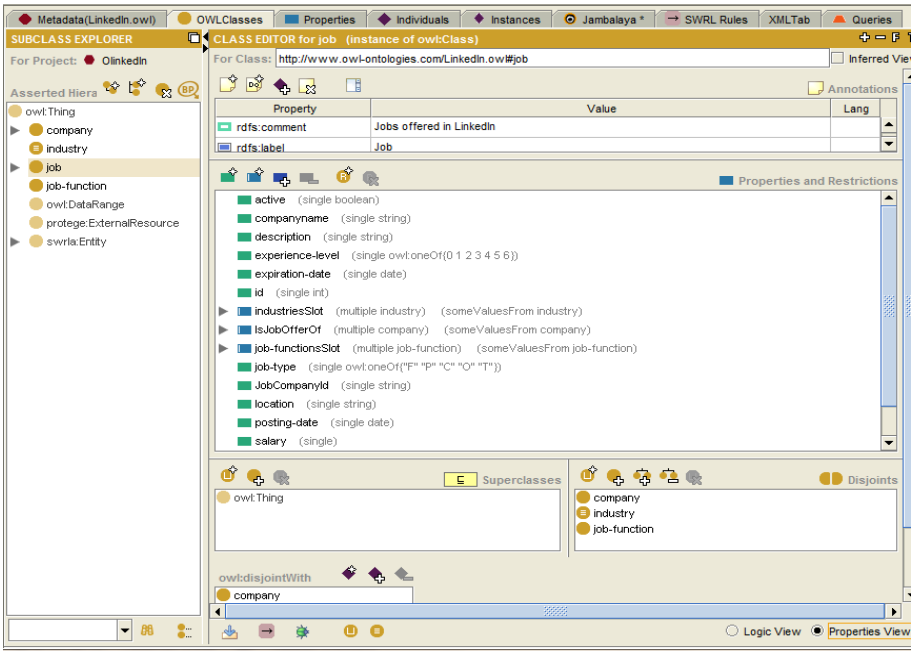
L'expressió per *PrivateCompany* queda:
(company-type has "P") or (company-type has "E") or (company-type has "O") or (company-type has "G")

L'expressió per *PublicCompany* queda:
(company-type has "Public_Company") or (company-type has "N") or (company-type has "D")

Job

Consta d'atributs bàsics com descripció, nom de la companyia, salari, activa, data de registre, etc. i d'altres més específics com tipus de feina, nivell d'experiència.

En la imatge es poden veure les propietats de la classe *job* el tipus de valor que admet cada una d'elles i la cardinalitat:



Imatge 9: Definició de la classe job

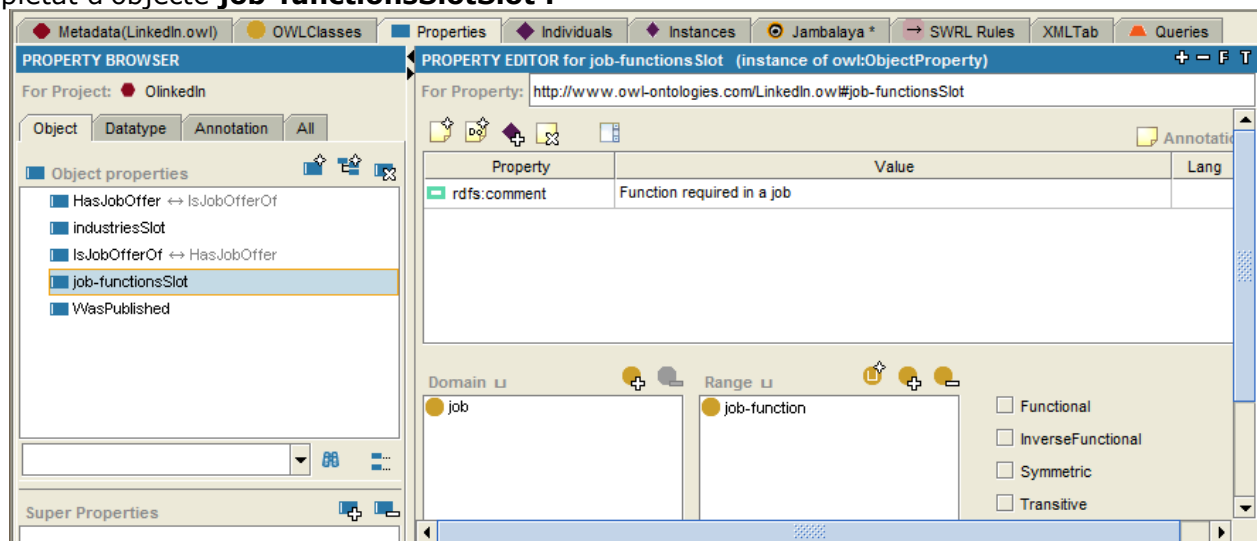
Per cada un dels atributs *job-type, experience-level* he definit el rang de valors que poden tenir en base a l' informació de l'estructura de Job a LinkedIn (veure apartat Enumeració de termes). Els he afegit com a atributs de classe en lloc de crear-les com a classes perquè en si no representen un concepte en concret, sinó que aporten redefinició per un concepte.

A més del les atributs de dades (DataType) també s'han afegit unes propietat d'objecte (ObjectProperty) per relacionar l'objecte o classe *job* amb la classe c*ompany*. Concretament es crea la propietat **IsJobOfferOf** (inversa de **HasJobOffer**) per trobar quina Companyia ofereix la feina.

Amb la intenció de permetre una classificació més detallada de cada individu (instància) de feines, he modelat unes subclasses dins de *job*. La taxonomia queda així:

- **Auditor** per classificar per funcions Accounting Auditing, Analyst, Consulting o Legal
- **BusinessManagement** per classificar per funcions Business Development, Finance, General Business o Management
- **Commercial** per classificar feines per funcions Sales, Marketing, Public Relations, Customer Service o Advertising
- **Engineer** per classificar feines per funcions Engineering o Information Technology
- **Educator** per classificar per funcions Education o Training
- **Industrial** per classificar per funcions Manufacturing, Production o Quality Assurance
- **ProductManagement** per classificar per funcions Product Management, Distribution, Supply chain o Purchasing
- **JobWithRequiredExperience** per classificar funcions que requereixen com a mínim un nivell d'experiència.

Per poder completar la definició d'aquestes subclasses, s'ha creat prèviament una propietat d'objecte **job-functionsSlotSlot :**



Imatge 10: propietat d'objecte job-functionsSlotSlot

Com es pot veure a l' imatge, aquest propietat aplica a la classe *job*, i per tant també a les seves subclasses *Engineer, Commercial, Educator, BusinessManagment, Industrial, ProductManagement i Auditor*. I el rang de valors el té sobre la classe *job-function*.

Per cada una de les subclasses de *job*, s'han especificat les expressions que indican els valors que ha de tenir la propietat *job-functionsSlot* per tal de que una oferta de feina sigui classificada dins d'una subclasse o un altre.

Auditor:

(job-functionsSlot has AccountingAuditing) or (job-functionsSlot has Consulting)

BusinessManagment:

(job-functionsSlot has GeneralBusiness) or (job-functionsSlot has BusinessDevelopment) or (job-functionsSlot has Management) or (job-functionsSlot has Finance)

Commercial:

(job-functionsSlot has CustomerService) or (job-functionsSlot has Sales) or (job-functionsSlot has Advertising) or (job-functionsSlot has Marketing) or (job-functionsSlot has PublicRelations)

Educator:

(job-functionsSlot has Education) or (job-functionsSlot has Training)

Engineer:

(job-functionsSlot has Engineering) or (job-functionsSlot has InformationTechnology)

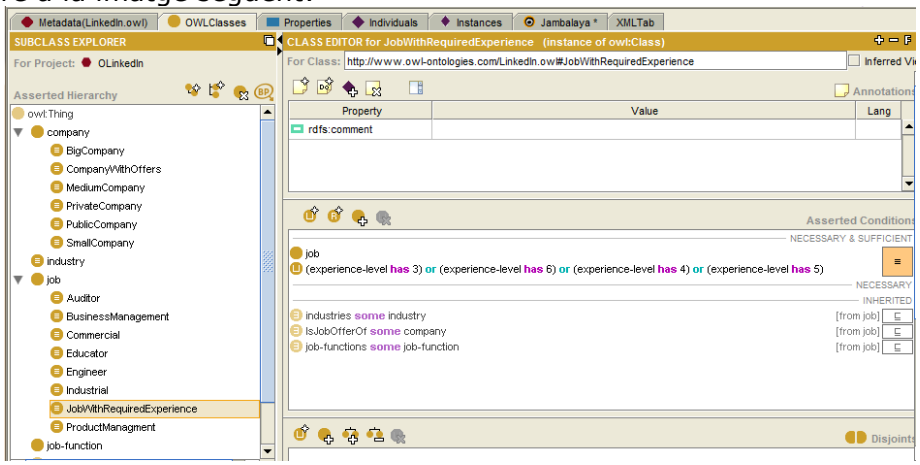
Industrial:

(job-functionsSlot has Manufacturing) or (job-functionsSlot has QualityAssurance) or (job-functionsSlot has Production)

ProductManagement:

(job-functionsSlot has SupplyChain) or (job-functionsSlot has Distribution) or (job-functionsSlot has Purchasing) or (job-functionsSlot has SupplyChain) or (job-functionsSlot has ProductManagementF) or (job-functionsSlot has StrategyPlanning)

Per definir la subclasse *JobWithRequiredExperience* s'ha aplicat una restricció sobre la propietat e*xperience-level*, en base els valors que pot tindre segons l'estructura de job a LinkedIn tal i com s'ha vist a l'apartat Enumeració de termes. L'expressió queda com es pot veure a la imatge següent:



Imatge 11: Asserció de la subclasse *JobWithRequiredExperience*

En la següent imatge es pot veure un diagrama, fet amb Jambalaya, de les associacions entre classes mitjançat les propietats d'objecte definides:

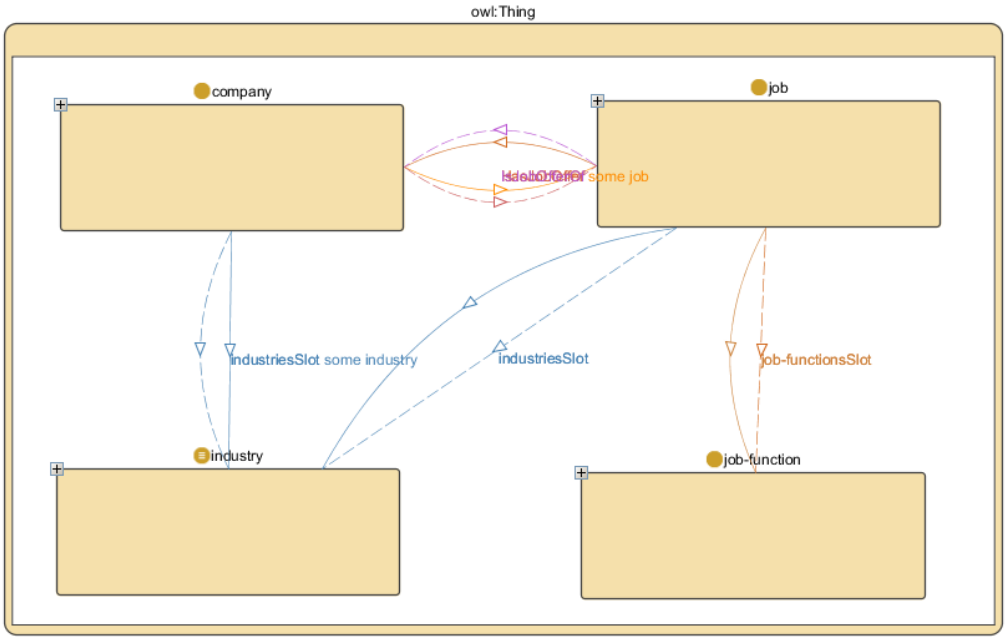


Diagrama 3: Diagrama de classes amb associacions per propietats

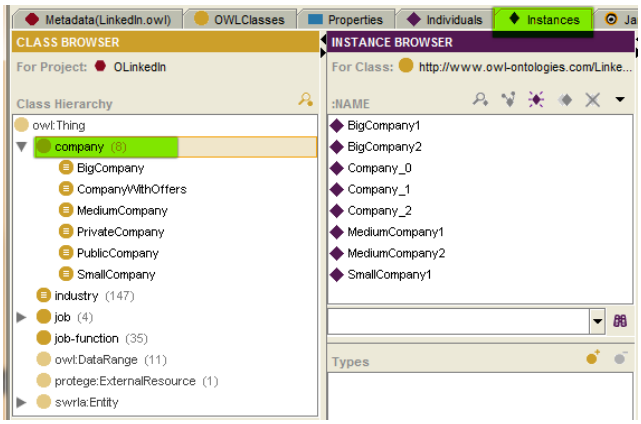
2.7. Instancies

Després d'haver implementat tota l'estructura de classificació a l'ontologia, he creat algunes instàncies individuals de les classes, de manera manual, a mode d'exemple per avaluar la consistència de les metadades.

Una instància es una realització específica de una classe. La instància d'una classe és un objecte. Es un membre d'una classe que té uns valors concrets per cada un dels seus atributs.

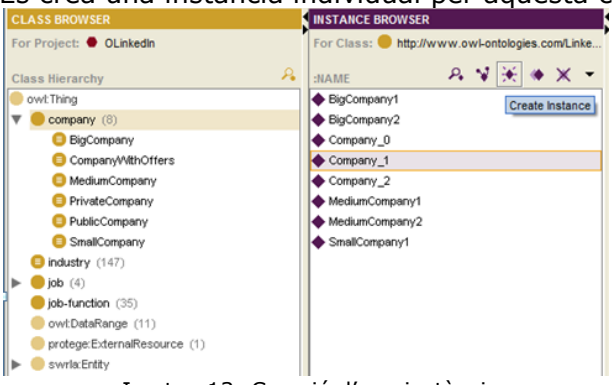
Per crear una instància individual per a una classe determinada es segueix el següent procés:

1. En la pestanya Instances, s'escull i es selecciona una classe:



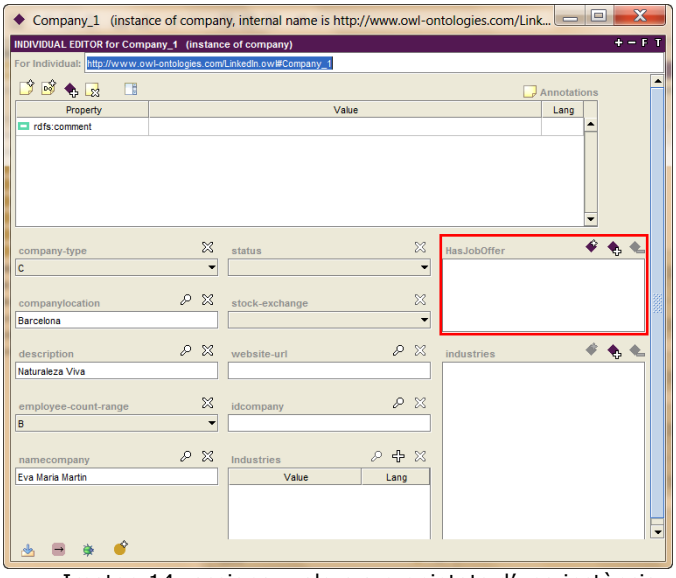
Imatge 12: Pestanya Instances

2. Es crea una instància individual per aquesta classe:



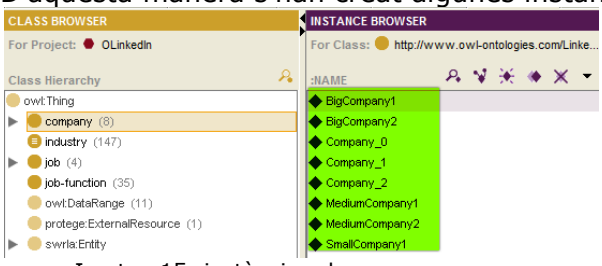
Imatge 13: Creació d'una instància

3. S'omplen els valors de les propietats:



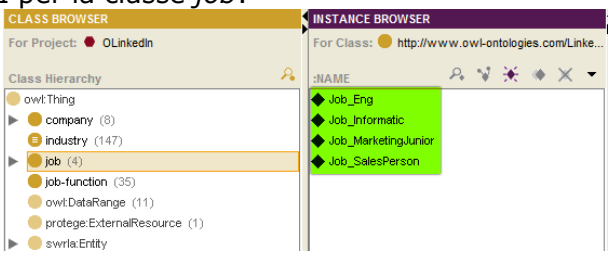
Imatge 14: assignar valors a propietats d'una instància

D'aquesta manera s'han creat algunes instàncies, per la classe *company*:



Imatge 15: instàncies classe company

I per la classe *job*:

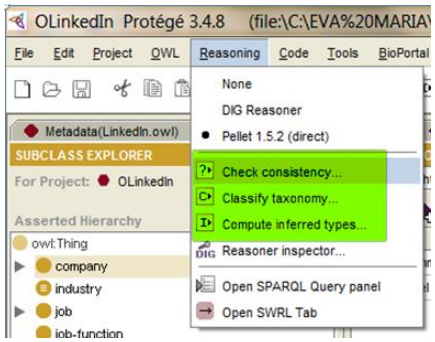


Imatge 16: instàncies classe job

2.8. Validació

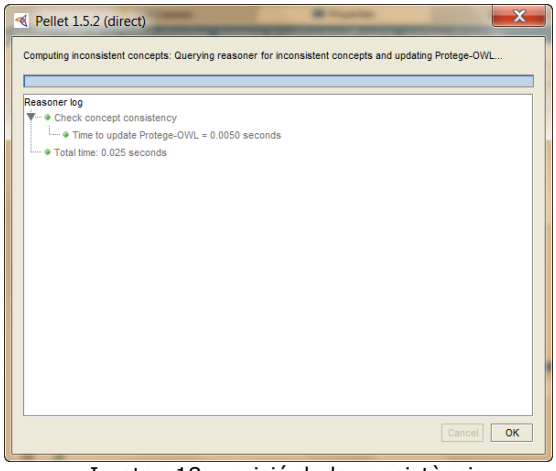
Les ontologies descrites amb subllenguatge OWL-DL permeten ser processades per un raonador o classificador. Un dels principals serveis que ofereix el raonador es la realització de les proves de subsumpció per tal de determinar la relació entre classes i subclasses i calcular així la jerarquia de l'ontologia. Un altre dels serveis estàndard que ofereix el raonador consisteix en comprovar, en base a la descripció (condicions) d'una classe si aquesta es inconsistent o no; serà inconsistent si no pot tenir alguna instància.

En la versió 3.4.8 de Protégé, està integrat el raonador Pellet 1.5.2. que permet validar alguns aspectes de la ontologia, entre ells: revisar la consistència de l'ontologia, obtenir automàticament la classificació taxonòmica i computar els tipus inferits (les instancies creades). L'activo i executo les accions, per ordre, que es veuen a la imatge:



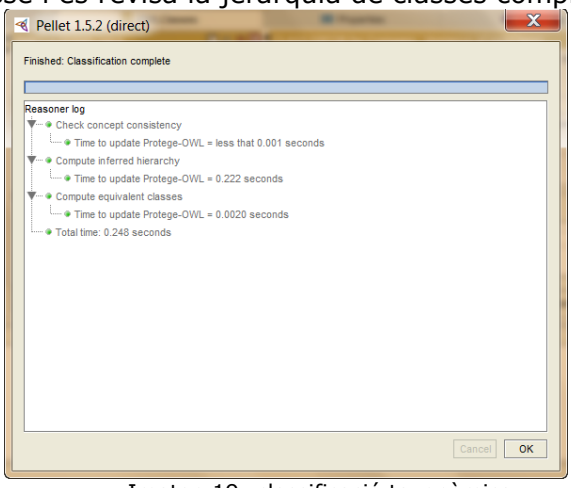
Imatge 17: opcions del raonador

Com a resultat de la revisió de la consistència obtinc el resultat:



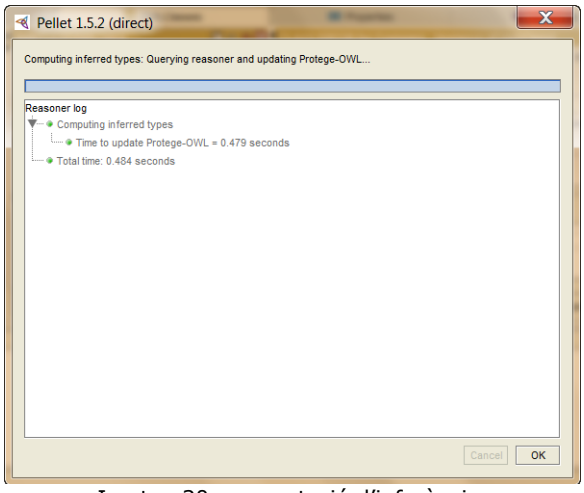
Imatge 18: revisió de la consistència

El resultat de l'execució de la classificació taxonòmica,on es valida la relació entre cada classe i es revisa la jerarquia de classes completa, és:



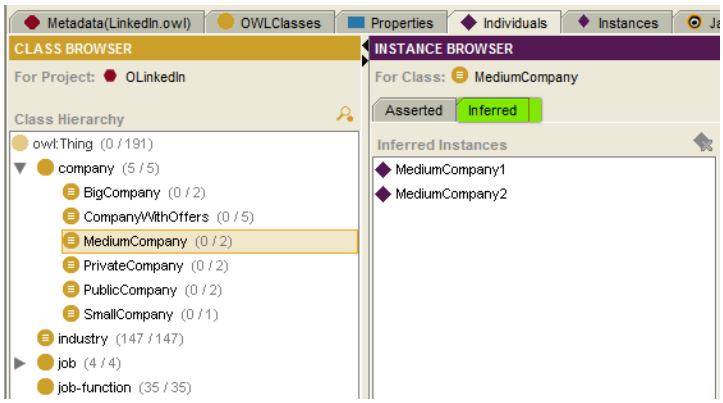
Imatge 19: classificació taxonòmica

En la computació del tipus inferits es troba les classes més específiques a les que pertany una instància, és a dir, es determina la classe a la que pertany cada un dels individus. El resultat obtingut ha estat:



Imatge 20: computació d'inferència

Es pot observar en la següent captura de pantalla com els individus han quedat classificats en cada classe (els números entre parèntesis al costat de cada classe ens indica el número de individus que han estat inferit a cada classe), concretament es veuen els individus de **MediumCompany:**



Imatge 21: inferència classe MediumCompany

3. Creació d'una aplicació per poblar l'ontologia

Com a segon objectiu d'aquest projecte, tal i com es presentava a l'apartat Objectius del TFC, es requereix un programa que permeti poblar l'ontologia creada a partir de les ofertes de feina (*Jobs*) i les empreses (*Companies*) de LinkedIn.

En els apartats que segueixen a continuació es definirà l'anàlisi, el disseny i el producte final per la creació d'aquesta aplicació.

3.1. Tecnologia

Començarem per una breu descripció de les diferents tecnologies utilitzades en el desenvolupament.

3.1.1. Microsoft Visual Studio[5]

Microsoft Visual Studio es un entorn de desenvolupament integrat (IDE, per les seves sigles en anglès) per a sistemes operatius Windows. Suporta múltiples llenguatges de programació com C++, C#, Visual Basic .NET, F#, Java, Python, Ruby, PHP; així com entorns de desenvolupament web com ASP.NET, Django, etc.

En aquest entorn, els desenvolupadors poden crear aplicacions que es comuniquen entre estacions de treball, pàgines web, així com serveis web, dispositius mòbils, etc. en qualsevol entorn que suporti plataforma .NET.

La introducció de la plataforma .NET de Microsoft va arribar amb la versió Visual Studio .NET (2002). .NET és una plataforma d'execució intermèdia multi llenguatge, de manera que els programes desenvolupats en .NET no es compilen en llenguatge màquina, si no en un llenguatge intermedi (CIL – CommonIntermediate Language) denominat Microsoft Intermediate Language (MSIL). El codi no es converteix a llenguatge màquina fins que es executat, per tant, el codi pot ser independent de la plataforma, l'únic requeriment és que la plataforma ha de tenir una implementació d'Infraestructura de llenguatge comú (CLI) per poder executar programes MSIL.

Aquesta versió Visual Studio .NET (2002) va representar una millora notable en la interfície, sent més neta i personalitzable. Va suposar també la introducció del llenguatge C#, un nou llenguatge dissenyat específicament per a la plataforma .NET, basat en C++ i Java.

3.1.2. C#[6]

C# és un llenguatge de programació orientat a objectes desenvolupat i estandarditzat per Microsoft com a part de la seva plataforma .NET, que després va ser aprovat com a estàndard per la ECMA (ECMA-334) i ISO(ISO/IEC 23270). C# és un dels llenguatges de programació dissenyats per la infraestructura de llenguatge comú.

La seva sintaxi bàsica deriva de C/C++ i utilitza el model d'objectes de la plataforma .NET, similar al de Java, però amb millores derivades d'altres llenguatges. Tot i que forma part de la plataforma .NET, C# es un llenguatge de programació independent.

3.1.3. OAuth[7]

OAuth (*Open Authorization*) és un protocol, proposat per Blaine Cook i Chris Messina, que permet autorització segura d'una API de mode estàndard i simple per a aplicacions d'escriptori, mòbils i web. La primera especificació formal, Oauth Core 1.0, va ser publicada en Octubre del 2007.

Es un mètode que permet a desenvolupadors de consumidors interactuar amb dades protegides i publicar-les; i per desenvolupadors de proveïdors de servei, proporciona als usuaris un accés a les seves dades protegint les credencials de la seva compte.

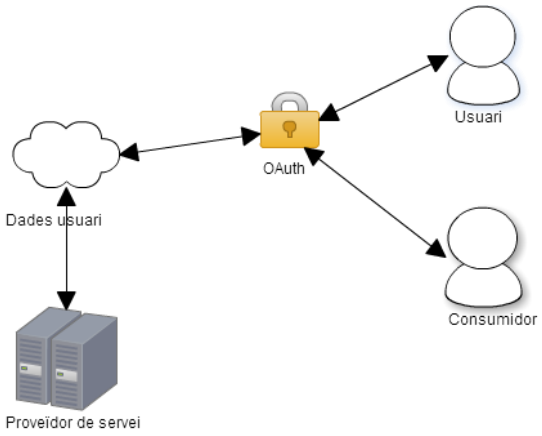


Diagrama 4: Autentificació OAuth

3.1.4. XML[8]

XML (eXtnsible Markup Language) és un llenguatge de marques desenvolupat per W3C (World Wide Web Consortium) que s'utilitza per emmagatzemar dades en forma llegible. És una proposta d'estàndard per l'intercanvio d'informació entre aplicacions independentment de quin sigui l'origen de les dades.

La seva estructura és senzilla , flexible i reutilitzable. Es tracta d'una estructura rigorosament jeràrquica en la que cada element està perfectament delimitat per unes etiquetes que el defineixen pel seu nom: <nom_element> element</nom_element>

Es important tenir en compte que XML és sensible a majúscules i minúscules.

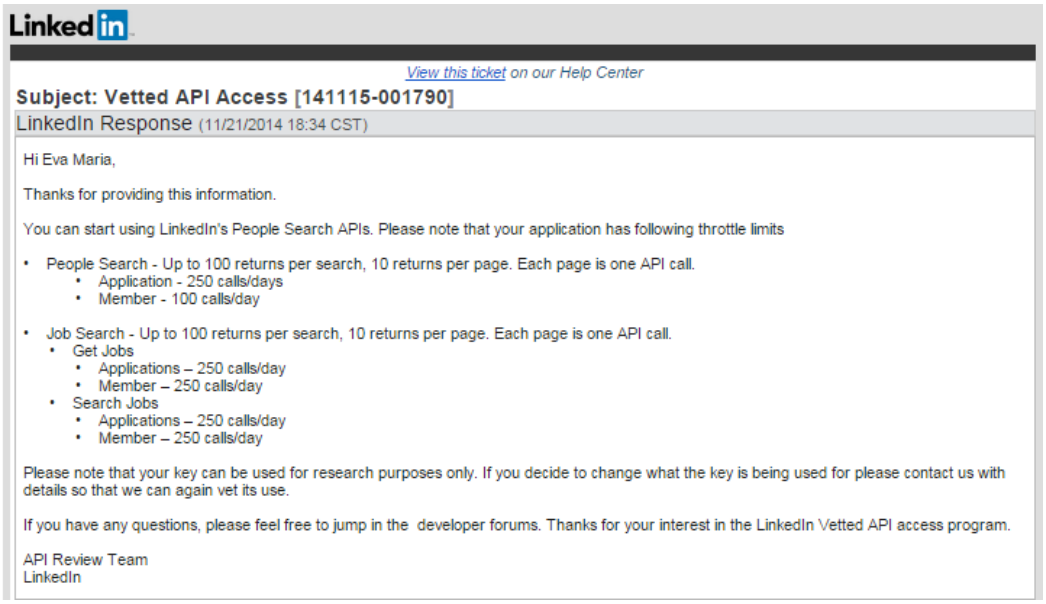
3.1.5. API LinkedIn

Una API (Interfícies de programació d'aplicacions) és un conjunt de funcions i procediments empaquetats en una biblioteca per poder ser utilitzats per altres programes com a capes d'abstracció.

LinkedIn ofereix una API per poder accedir a les dades del seu domini. Però per poder tenir accés a aquesta API, es requereix prèviament registrar l'aplicació per la que el desenvolupador vol fer-la servir i demanar autorització. Llavors LinkedIn avalua l'aplicació i l'objectiu de la mateixa per determinar si dona autorització o no.

Es por sol·licitar clau d'accés a l'API de LinkdeIn mitjançant l'enllaç: https://www.linkedin.com/secure/developer?newapp=

Una vegada acceptada la sol·licitud, l'equip de Revisió API et comunica de les teves clau d'accés i de quan pots començar a fer-les servir:



LinkedIn

View this ticket on our Help Center

Subject: Vetted API Access [141115-001790]

LinkedIn Response (11/21/2014 18:34 CST)

Hi Eva Maria,

Thanks for providing this information.

You can start using LinkedIn's People Search APIs. Please note that your application has following throttle limits

- People Search - Up to 100 returns per search, 10 returns per page. Each page is one API call.
  - Application - 250 calls/days
  - Member - 100 calls/day
- Job Search - Up to 100 returns per search, 10 returns per page. Each page is one API call.
  - Get Jobs
    - Applications – 250 calls/day
    - Member – 250 calls/day
  - Search Jobs
    - Applications – 250 calls/day
    - Member – 250 calls/day

Please note that your key can be used for research purposes only. If you decide to change what the key is being used for please contact us with details so that we can again vet its use.

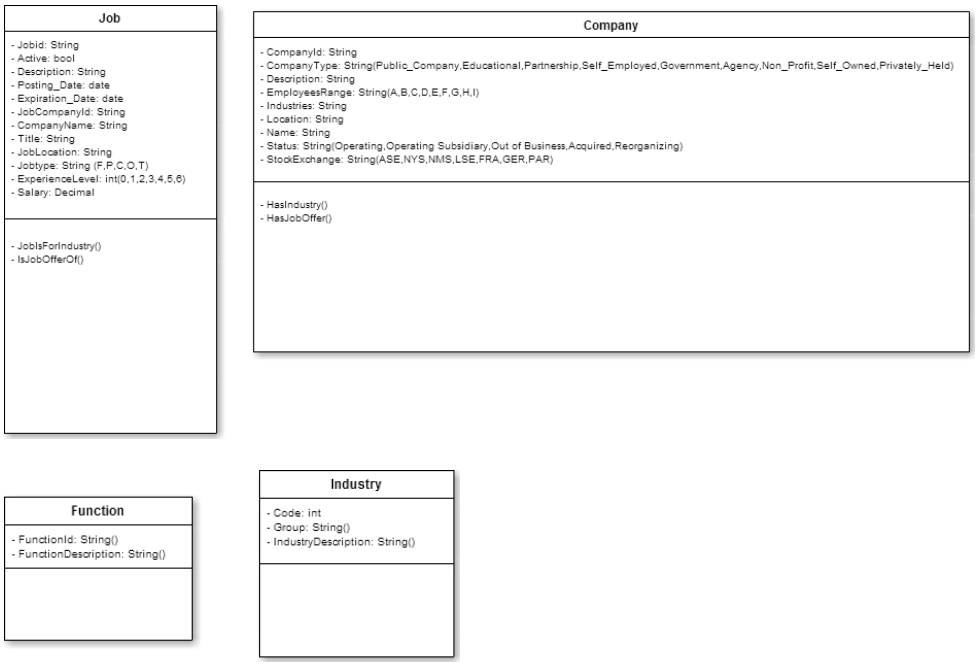
If you have any questions, please feel free to jump in the developer forums. Thanks for your interest in the LinkedIn Vetted API access program.

API Review Team
LinkedIn

3.2. Anàlisi de l'aplicació

L'objectiu funcional de la aplicació consisteix en obtenir informació d'ofertes de feina i companyies existents al domini de LinkedIn. En l'Apartat enumeració de termes ja mostro quina es l'estructura d'aquestes classes a LinkedIn, i sobre aquesta estructura s'ha construït l'ontologia.

En els següents diagrames UML es por veure la definició de les classes:



S'ha creat una aplicació en C# sobre plataforma Microsoft Visual Studio 2013. Aquesta aplicació , mitjançant la API que ofereix LinkedIn als desenvolupadors, descarrega informació de Job i Company, la processa i genera un fitxer job.xml i un altre companies.xml que contenen única i exclusivament la informació que volem per omplir amb dades cada un dels atributs de les classes de la nostra ontologia, és a dir, poblar l'ontologia d'instàncies.

Aquests fitxers xml, els podem importar a l'ontologia que ja tenim creada fent servir l'eina XMLTab que ofereix Protégé 3.4.

XMLTab permet als usuaris importar un document XML en P Protégé 3.4., creant un conjunt de classes (en cas de no trobar la classe amb el mateix nom de l'etiqueta en l'estructura del fitxer xml que s'està important) i instàncies d'una ontologia que corresponen a les entrades en el document XML. També permet generar una estructura per una ontologia.

Per a la creació de l'aplicació, s'ha considerat també, que la plataforma LinkedIn està viva, és susceptible de canvis que poden afectar directament a l'estructura de Job i Company i, per tant, el nostre programa pot esdevenir obsolet. Es per això que el muntatge del programa permet una fàcil actualització del parser (analitzador sintàctic) de l'estructura xml que es rep de LinkedIn en cas de ser requerit en el futur.

Pel que fa a l'ontologia, en cas de que l'estructura de les classes en el domini LinkedIn esdevingués molt diferent, hauríem de descartar la ontologia creada i partir de cero fent una importació amb el XMLTab, sobre una nova ontologia en blanc, dels fitxers xml generats pel programa; aquesta acció crearia automàticament les classes, atributs i instàncies en la mateixa importació; després de manera manual s'haurien de definir les restriccions . Si els canvis esdevinguts en l'estructura a LinkedIn amb el pas del temps són mínims, llavors es pot adaptar l'ontologia OLinkedIn manualment.

3.3. Implementació

He donat el nom **LinkedInAPIConsumer** a l'aplicació.

Per al disseny i desenvolupament del programa, s'han analitzat i reutilitzat parts de codis, principalment, d'exemples i especificacions disponibles i/o accessibles des de la web de desenvolupadors de LinkdeIn:
https://developers.linkedin.com/documents/libraries-and-tools

El funcionament de l'aplicació permet la connexió a l'API del web site de LinkedIn, l'accés a les dades de Job i Company, la descàrrega d'aquestes dades i la seva transformació en uns fitxers xml compatibles amb la funcionalitat d'importació xml de Protégé.

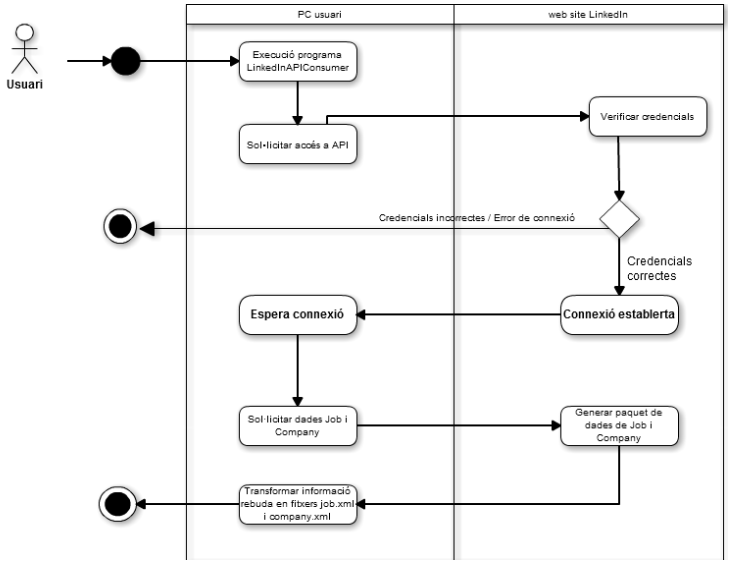
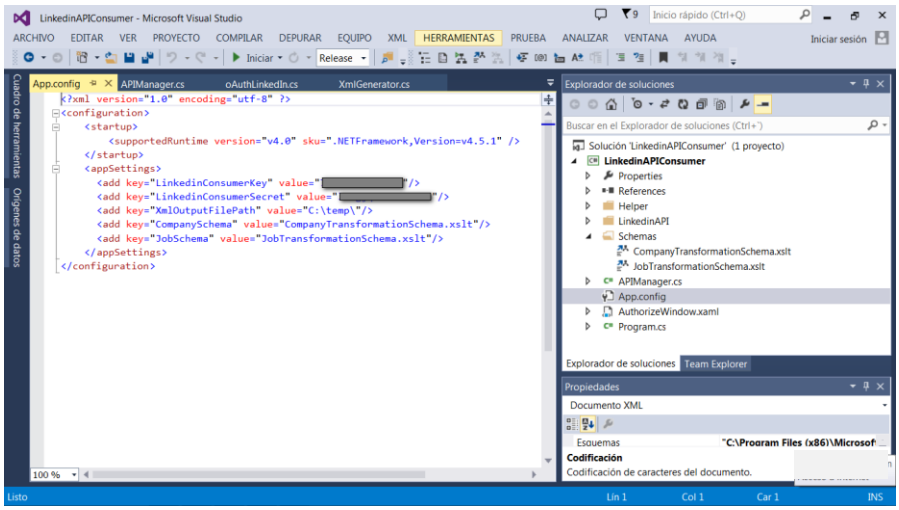
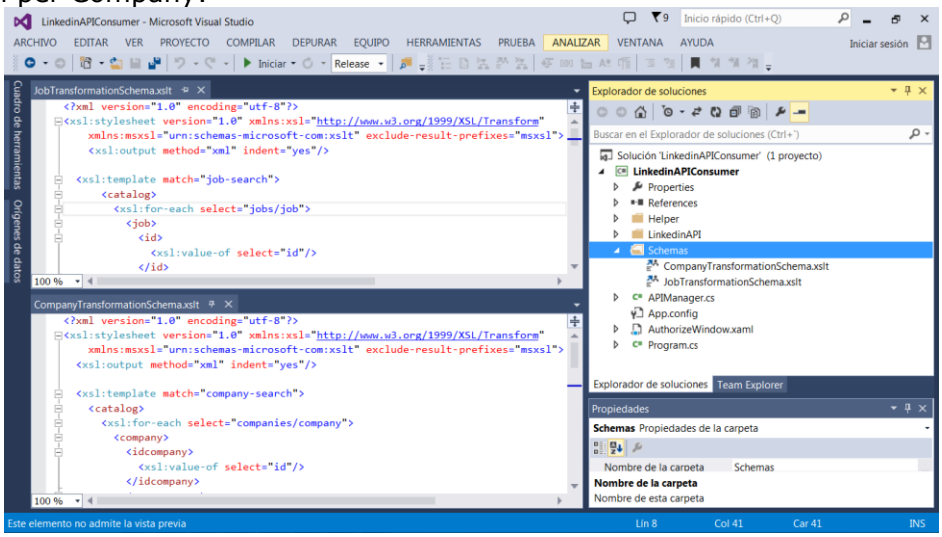


Diagrama 5: Diagrama d'estats de l'aplicació LinkedInAPIConsumer

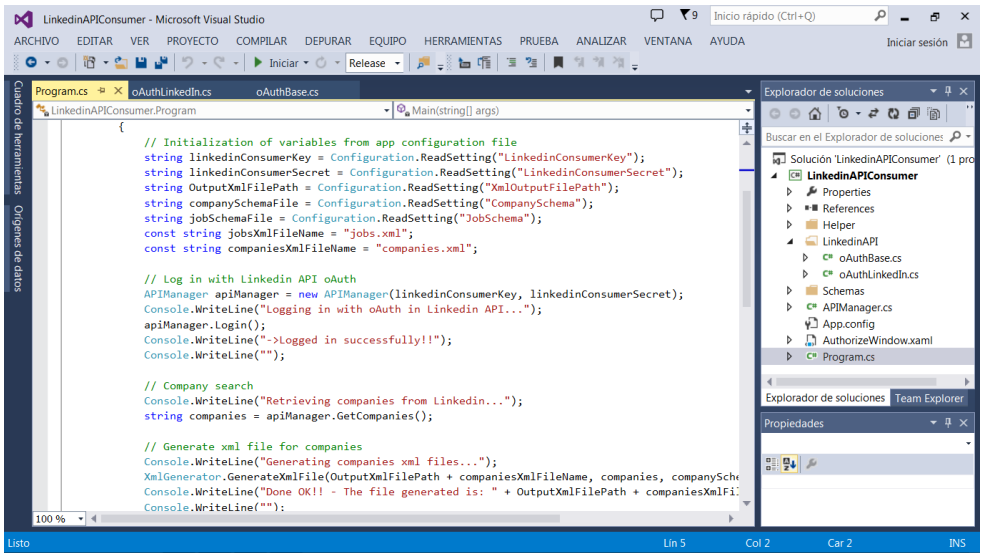
Com comentava en l'apartat d'anàlisi, hi ha una part de l'aplicació que s'ha de permetre modificar fàcilment, es per això que en un fitxer a part, App.config, he definit les variables per emmagatzemar les dades de clau d'accés i codi secret (atorgats per LinkedIn), la ruta per desar els fitxers job.xml i companies.xml, i els esquemes que farem servir per a la construcció d'aquets fitxers:



En una carpeta a part, he afegit les classes encarregades de transformar la informació obtinguda de LinkedIn per generar els XML amb la estructura que desitgem tant per Job com per Company:

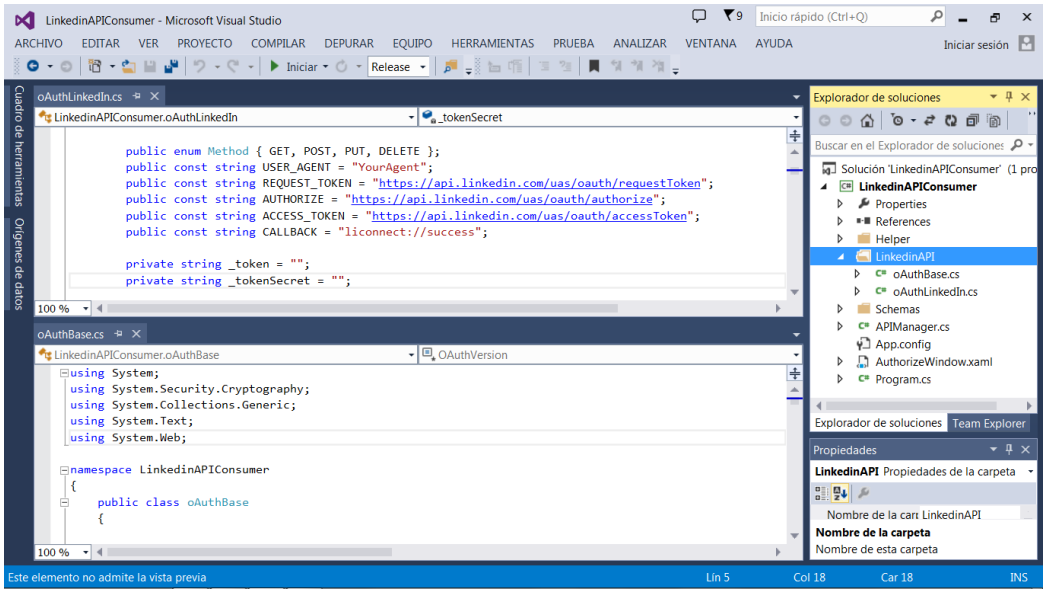


La classe principal d'execució del programa (Program.cs) està documentada de manera que es pot seguir pas a pas les crides que es van fent d'inicialització de variables, accés a LinkedIn, extracció de dades de Company, generació del XML amb la informació de Company i finalment el mateix per Job.



3.3.1. Procés de accés a l'API de Linkedin

Es creen unes classes per realitzar la validació de credencials, generació del token i connexió a l'API de LinkedIn



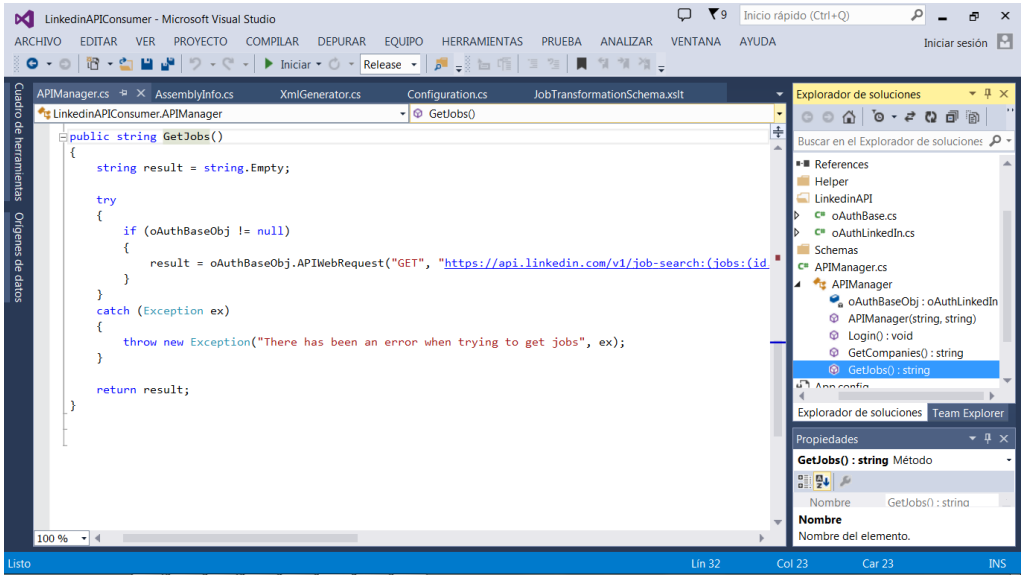
3.3.2. Extracció de dades d'ofertes de feina (Job)

Per a l'extracció de dades de Job es fa servir la sentència:

```
result = oAuthBaseObj.APIWebRequest("GET", "https://api.linkedin.com/v1/job-search:(jobs:(id,active,posting-date,expiration-date,company:(id,name),position:(title,location,job-functionsSlot,industries,job-type,experience-level),salary))", null);
```

Es tracta d'una consulta de tipus sol·licitud, "GET", dirigida a l'API de LinkedIn per la recerca d'ofertes de feina https://api.linkedin.com/v1/job-search. Entre parèntesis s'especifiquen els camps dels quals es vol rebre informació per cada oferta de feina, és a dir, es sol·liciten els següents atributs:

- *id, active*: identificador de la feina i estat de la seva vigència.
- *posting-date*: data de publicació de la oferta de feina.
- *expiration-date*: data de expiració de l'oferta de feina.
- *id* i *name* de la companyia: identificador i nom de la companyia que ofereix la feina publicada.
- *title*: posició per a la que s'ofereix la feina.
- *job-functionsSlot* i *job-type*: àrees funcionals en les que es classifica la oferta de feina i tipus de contracte que s'ofereix
- *industries*: sectors als que pertany la companyia que ofereix la feina.
- *experience-level*: nivell d'experiència que es requereix en l'oferta.
- *Location:* ubicació on s'ofereix la feina.
- *salary*:salari que s'ofereix.



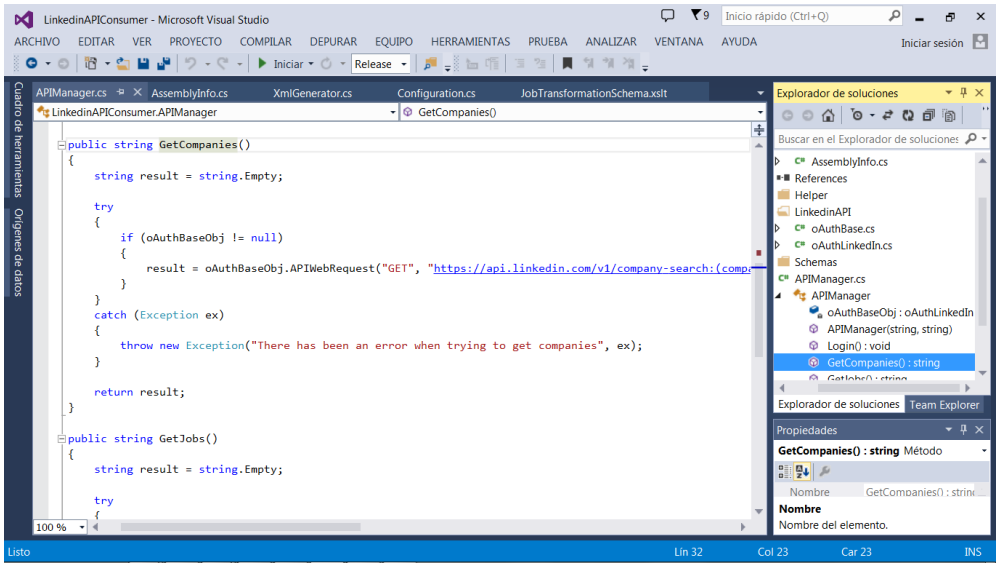
3.3.3. Extracció de dades de Companyies (Company)

Per l'extracció de dades de Company es fa servir la sentència de consulta:

```
result = oAuthBaseObj.APIWebRequest("GET", "https://api.linkedin.com/v1/company-search:(companies:(id,name,company-type,website-url,industries,status,employee-count-range,locations,description,stock-exchange))?keywords=", null);
```

Es tracta d'una consulta de tipus sol·licitud, "GET", dirigida a l'API de LinkedIn per la recerca de companyies https://api.linkedin.com/v1/company-search. Entre parèntesis s'especifiquen els camps dels quals es vol rebre informació per cada companyia, és a dir, es sol·liciten els següents atributs:

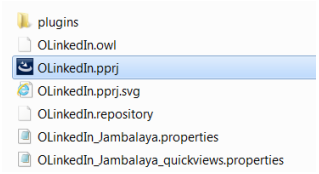
- *id i name*: identifiador i nom de la companyia.
- *company-type*: tipus de companyia, pública, privada, governamental, etc.
- *website-url*:direcció web de la companyia.
- *industries*: sectors empresarials a la que pertany la companyia.
- *status*:estat en el que es troba la companyia, operativa, en fallida, etc.
- *employee-count-range*: nombre d'empleats que hi treballen a la companyia.
- *locations*:presència geogràfica de la companyia.
- *description*: descripció.
- *stock-exchange*:borsa en la que cotitza la companyia



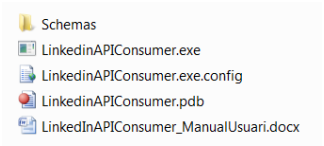
4. Paquet de la solució

El lliurament a l'usuari consta de dos paquets:

- **OLinkedin.rar:** aquest conté l'ontologia i els seus fitxers relacionats.



- **Release.rar:** aquest conté l'aplicació per l'extracció de les dades des de la plataforma LinkedIn, i el manual per l'usuari.



4.1. Manual d'usuari

LinkedInAPIConsumer és una aplicació dissenyada per realitzar extraccions automàtiques de dades d'ofertes de feina i companyies dins del domini de LinkedIn amb l'objectiu de ser utilitzades per poblar la ontologia OLinkedIn.

Aquest manual descriu no només com executar la aplicació, sinó que també explica el procés d'importació a la ontologia de la informació obtinguda.

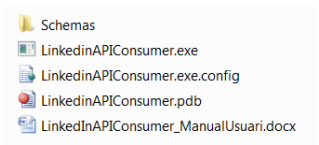
4.2. Requeriments

L'usuari ha de estar registrar al domini de LinkedIn, és requisit imprescindible que inicïi sessió amb una compte vàlida.

Sistema operatiu Windows 7 o superior
Protégé Versió 3.4.8 (Build 629). Es pot descarregar lliurament a http://protege.stanford.edu/ on trobarà també les instruccions d'instal·lació

4.3. Instal·lació

Crear una carpeta, per exemple amb el nom LinkedInAPIConsumer i descomprimir el fitxer Release.rar en aquesta carpeta.



Descomprimir el fitxer OLinkedIn.rar.

4.4. Configuració

El fitxer ***LinkedinAPIConsumer.exe.config*** inclou els paràmetres necessaris pel correcte funcionament de l'aplicació i que poden modificar-se segons necessitats de l'usuari:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
    <startup>
        <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.5.1" />
    </startup>
    <appSettings>
      <add key="LinkedinConsumerKey" value="XXXXXXXXXXXXXXX"/>
      <add key="LinkedinConsumerSecret" value="XXXXXXXXXXXX"/>
      <add key="XmlOutputFilePath" value="C:\temp\"/>
      <add key="CompanySchema" value="CompanyTransformationSchema.xslt"/>
      <add key="JobSchema" value="JobTransformationSchema.xslt"/>
    </appSettings>
</configuration>
```

LinkedinConsumerKey és la clau d'accés atorgada per LinkedIn. Si no té una altre clau vàlida, no modifiqui aquest paràmetre.

LinkedinConsumerSecret és la clau secreta que acompanya a la clau d'accés, atorgada també per LinkedIn. . Si no té una altre clau vàlida, no modifiqui aquest paràmetre.

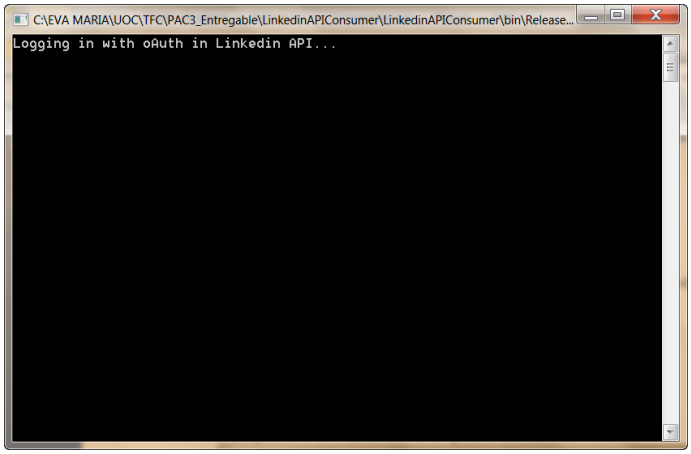
XmlOutputFilePath és ruta on es vol que el programa desi el fitxers XML que genera l'aplicació amb les dades extretes de LinkedIn.

CompanySchema i **JobSchema** són els esquemes corresponents per els parses generadors dels fitxers XML de Company i Job respectivament en el format desitjat.

4.5. Extracció de dades de LinkedIn

Executar el programa ***LinkedinAPIConsumer.exe*** que trobarà a la carpeta Release fent doble clic sobre el fitxer amb el ratolí. Pot crear un accés directe i col·locar-lo al Escriptori o al menú d'inici per més comoditat en futures execucions del programa.

Primerament s'obrirà una pantalla MS-DOS informant de que s'està fent una connexió a l'API de LinkedIn:

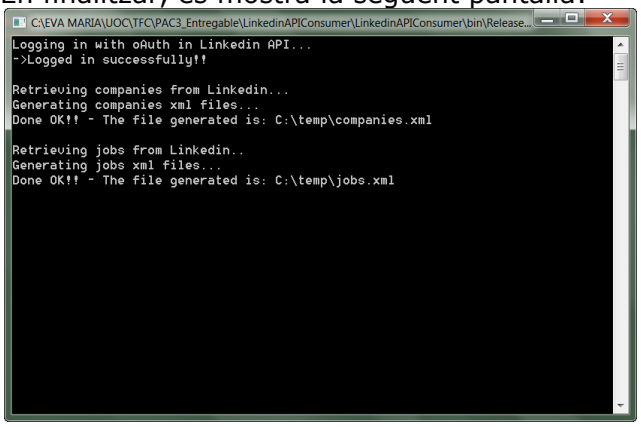


I immediatament es sol·licita el usuari i contrasenya per accedir:

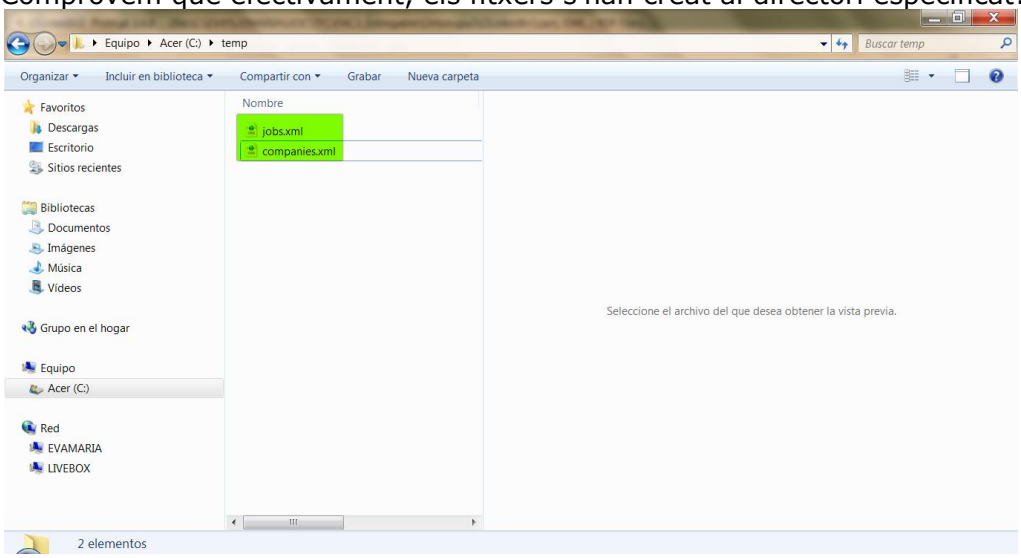
L'usuari ha d'iniciar sessió amb la seva compte personal a LinkedIn.

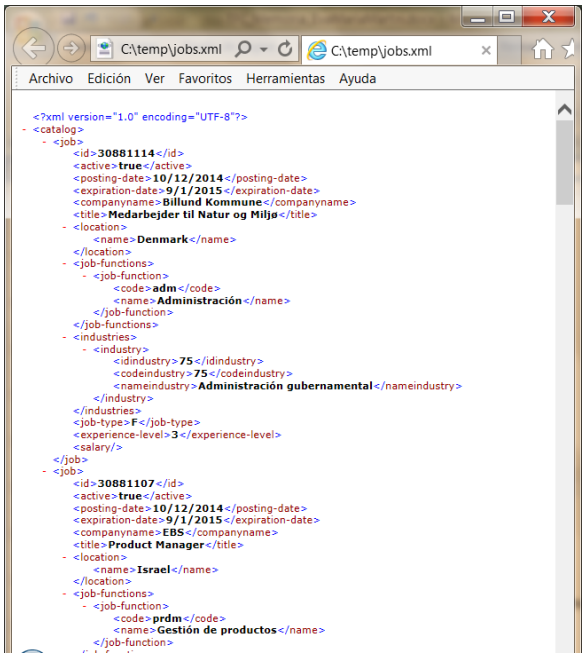
Polsar el botó "Permitir acceso" i llavors el sistema validarà les dades d'accés (usuari, contrasenya, clau d'accés a l'API). A través de oAuth, el sistema genera i atorga un token vàlid permet la connexió durant un temps determinat de temps. Si tot es correcte el programa podrà extreure les dades i generar els fitxers XML.

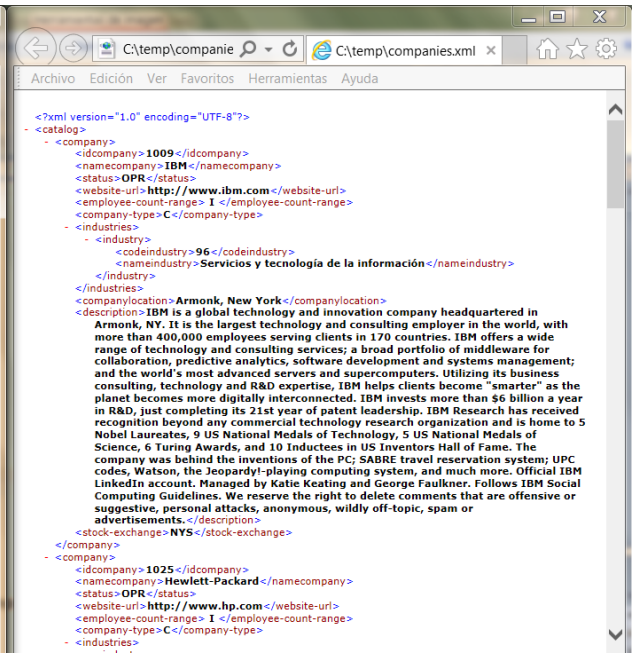
En finalitzar, es mostra la següent pantalla:



Comprovem que efectivament, els fitxers s'han creat al directori especificat:



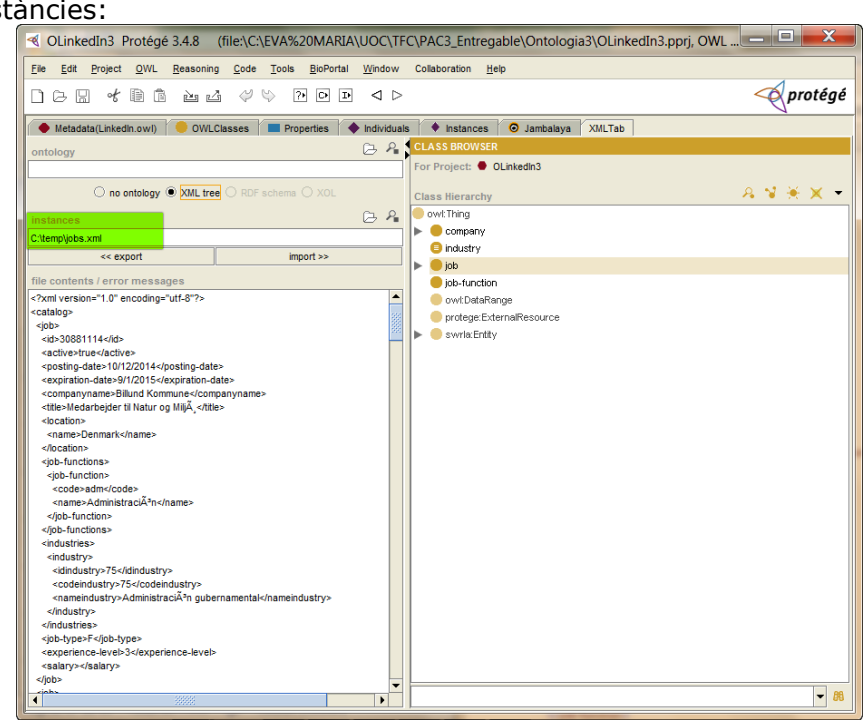




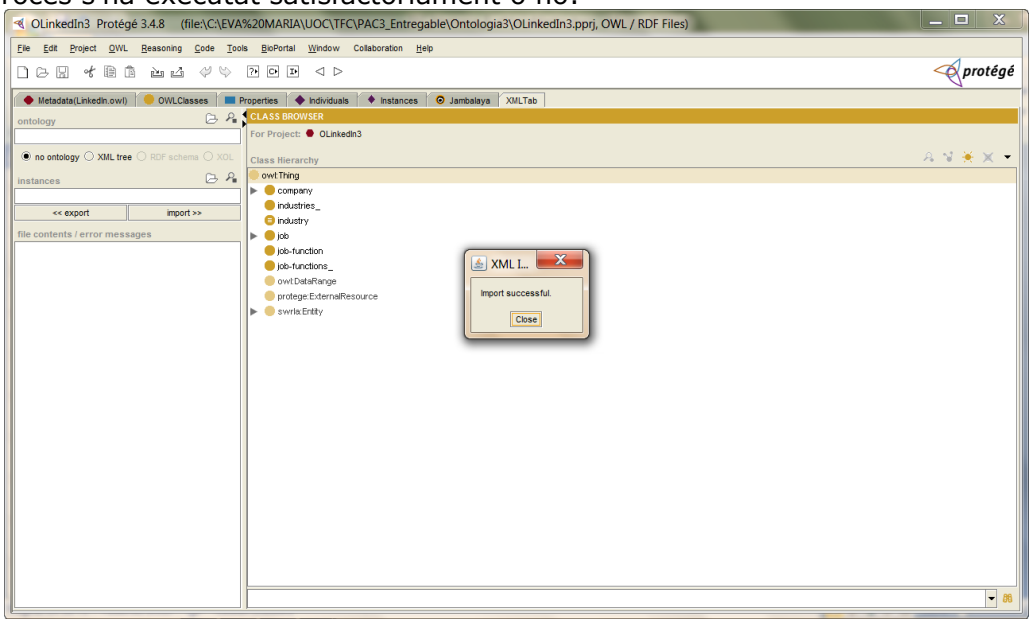
4.6. Importació de dades a l'ontologia. Creació d'instàncies.

Accedim al programa Protégé i obrim l'ontologia OlinkedIn que es troba a la carpeta Ontologia (fitxer ***OLinkedIn.pprj***)

Anem a la pestanya "XMLTab" i indiquem el fitxer XML que es vol importar per crear instàncies:



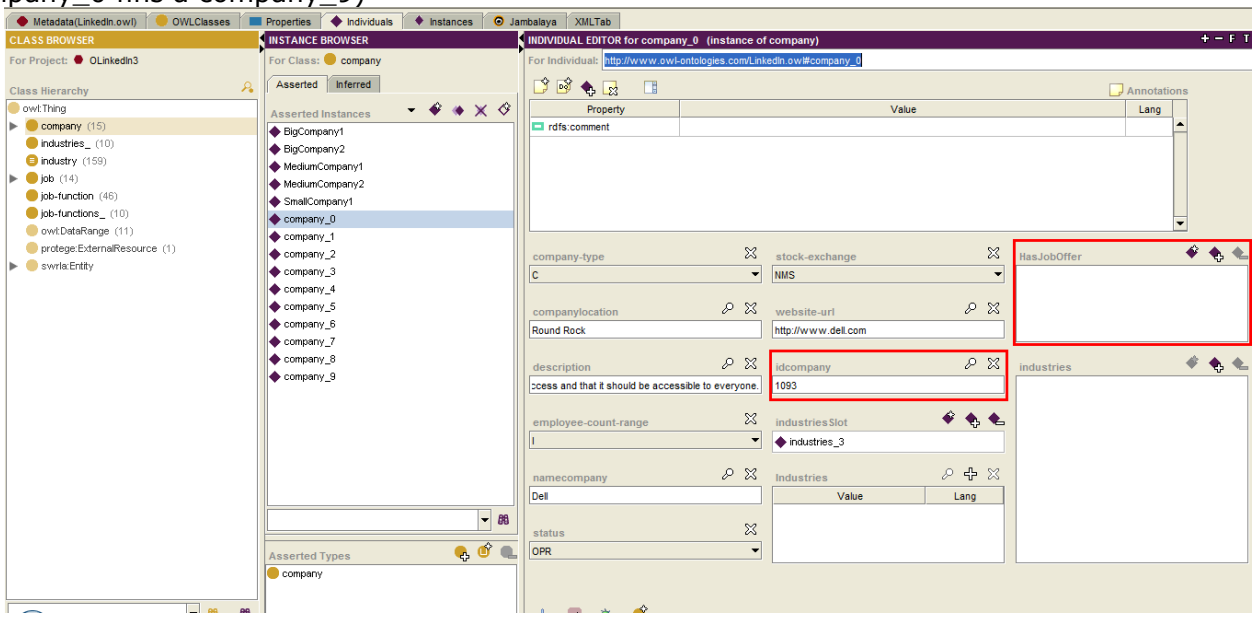
En aquesta pantalla es polsa el botó Import, i en finalitzar el procés el sistema informa si el procés s'ha executat satisfactòriament o no:



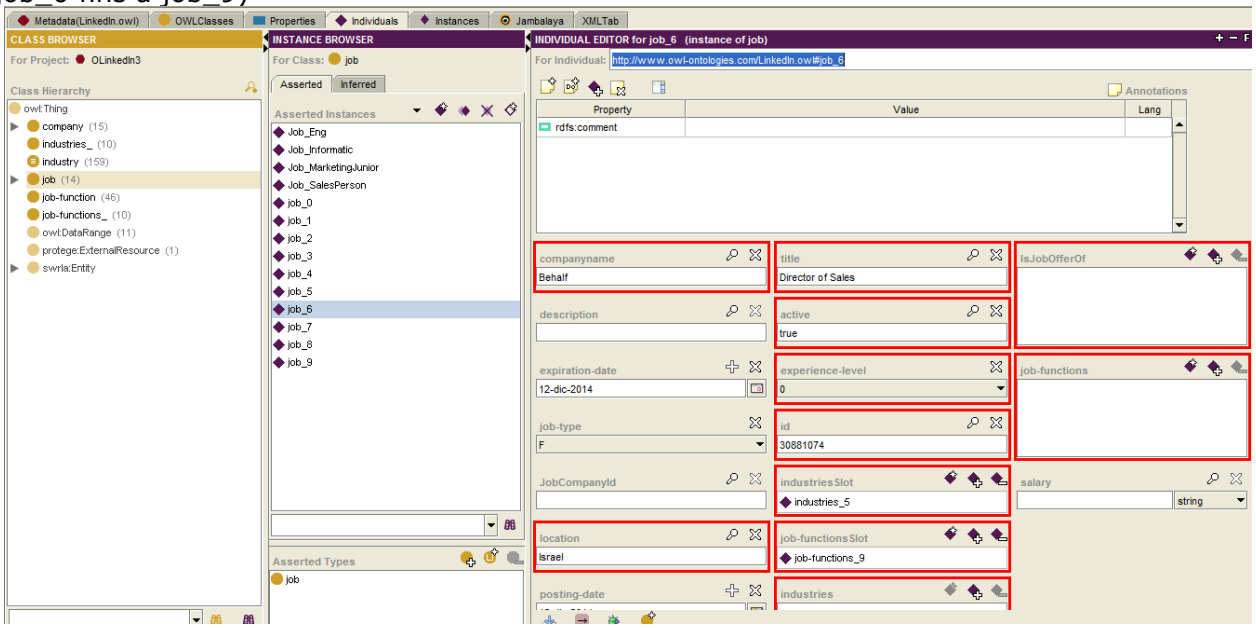
5. Proves

Tal i com s'ha descrit al manual d'usuari, fem una extracció de dades de Job i Company i provem d'importar-les a la ontologia. El resultat obtingut és la creació automàtica d'instàncies dins de les classes definides a OLinkedIn.

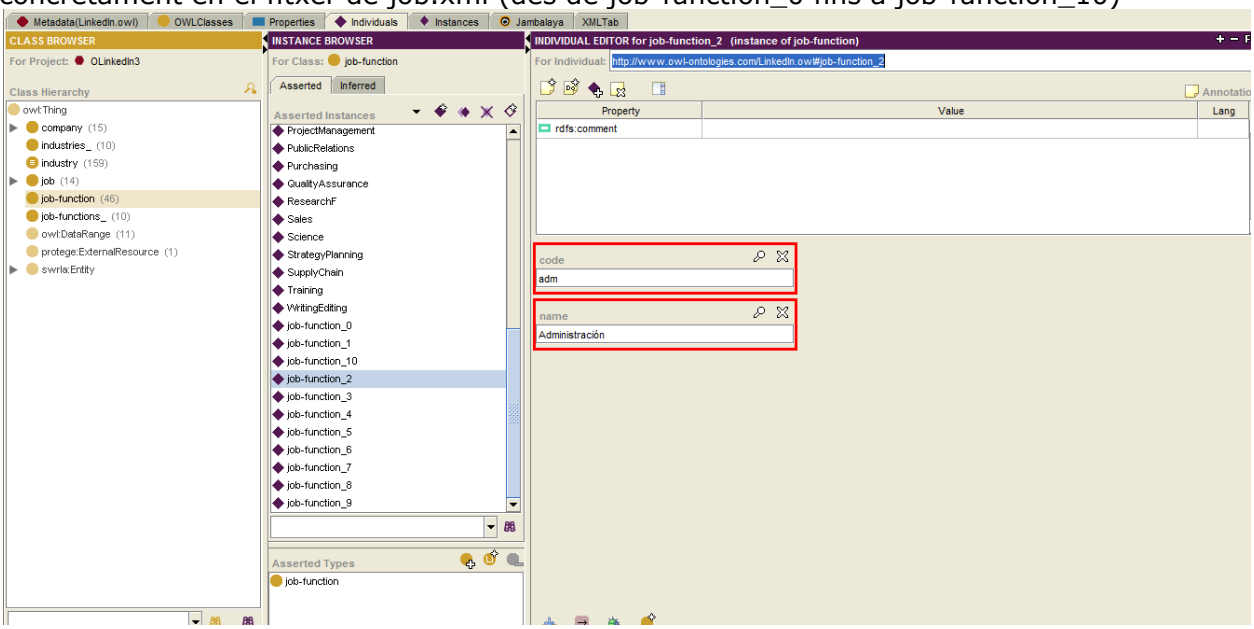
S'han creat instàncies de les 10 companyies rebudes de la consulta a LinkedIn(des de company_0 fins a company_9)



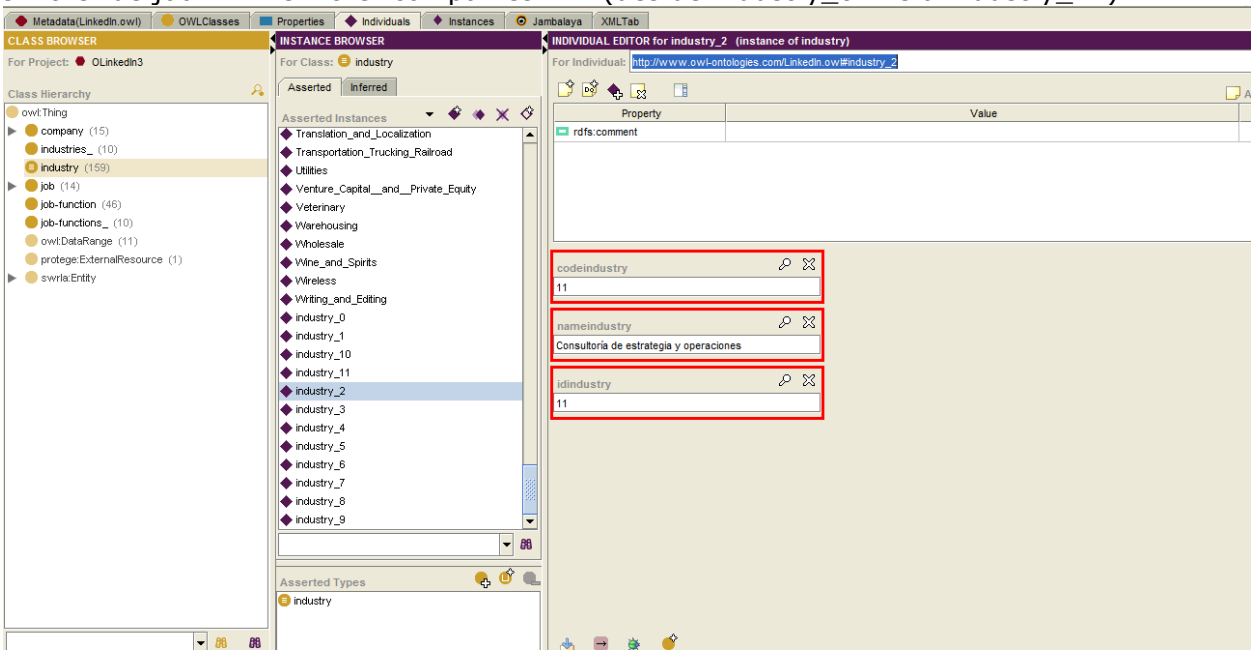
S'han creat instàncies de les 10 ofertes de feina rebudes de la consulta a LinkedIn(des de job_0 fins a job_9)



S'han creat instàncies de 11 job-function rebudes de la consulta a LinkedIn, concretament en el fitxer de job.xml (des de job-function_0 fins a job-function_10)



S'han creat instàncies de 12 industry rebudes de la consulta a LinkedIn, repartides entre el fitxer de job.xml i el fitxer companies.xml (des de industry_0 fins a industry_11)



El procés d'importació ha creat també dos noves classes *industries* i *job-funtions* per emmagatzemar els valors de les ObjectProperty *industriesSlot* i *job-functionSlot* de cada instància de *company* i *job*.

6. Conclusions

6.1. Introducció

En aquest últim apartat faré referència als coneixements adquirits amb la realització d'aquest treball, un anàlisi crític del seguiment de la planificació i metodologia establerts per a la realització del treball, i una reflexió sobre l'assoliment dels objectius plantejats inicialment.

6.2. Valoració

Per la realització d'aquest projecte ha estat necessari l'estudi de conceptes i tecnologies de les que no en tenia coneixement.

La creació de l'ontologia va ser el primer repte. Fer un estudi dels passos a seguir, dels components i conèixer el funcionament de l'editor Protégé. L'objectiu ha estat assolit sense problemes.

Després, la creació de l'aplicació per l'extracció de dades, segon objectiu del projecte va comportar més temps del planificat i certes complicacions. Primerament van sorgir alguns problemes en la sol·licitud de credencials per l'accés a l'API, va ser denegada en primera instància i es van perdre dies fins a aconseguir la clau d'accés. Després, la programació en C# va implicar moltes hores de programació i ajustaments a la ontologia.

Aquest projecte m'ha despertat un gran interès pel món de la Web Semàntica i les ontologies , i sens dubte continuaré profunditzat en la matèria.

6.3. Línees futures

Com a línees futures immediates de treball, un primer objectiu seria profunditzar en l'estudi de SPARQL, un llenguatge de consulta de conjunts de dades RDF, per arribar a crear un programa o un conjunt de regles sobre l'ontologia que permeti analitzar i fer consultes sobre la informació de les ofertes de feina.

Després millorar la implementació de la aplicació per aconseguir que el procés d'importació de dades en la ontologia no creï les dos classes *industries* i *job-functions*. Estudiar la versió més actual de Protégé per veure si aporta noves funcionalitats o resolucions compatibles amb la ontologia OLinkedIn.

Pel que fa al manteniment de l'ontologia, en principi el manteniment de l'ontologia serà a càrrec meu. Periòdicament revisaré les actualitzacions sobre l'estructura dels registres Job i Company a la web de LinkedIn per tal de comprovar si s'ha d'afegir o extreure informació a l'Ontologia; i per tant també fer els ajustos pertinents a l'aplicació d'importació des de LinkedIn cap a l'Ontologia. De totes maneres, qualsevol usuari consumidor de l'ontologia *OlinkedIn* pot fer manteniment de la mateix amb un editor d'Ontologia compatible amb OWL DL.

7. Glossari

Infoxicació: Trastorn intel·lectual producte de la incapacitat d'analitzar i comprendre una pluja d'informació com la que poden proporcionar els mitjans electrònics actuals.

8. Bibliografia

8.1. Tutorials

Matthew Horridge1, Holger Knublauch2, Alan Rector1, Robert Stevens1, Chris Wroe1 (August 27, 2004). **A Practical Guide To Building OWL Ontologies Using The Protégé-OWL Plugin and CO-ODE Tools.** Edition 1.0

1 The University Of Manchester
2 Stanford University

8.2. Internet

1. Berners-Lee
   http://www.w3.org/People/Berners-Lee/

2. RDF
   http://www.w3.org/RDF/

3. Protégé
   http://protegewiki.stanford.edu/wiki/Main_Page
   http://protege.stanford.edu/

4. OWL
   http://www.w3.org/2001/sw/wiki/OWL
   http://www.w3.org/2007/09/OWL-Overview-es.htm

5. Microsoft Visual Studio
   http://es.wikipedia.org/wiki/Microsoft_Visual_Studio

6. C#
   http://es.wikipedia.org/wiki/C_Sharp

7. OAuth
   http://es.wikipedia.org/wiki/OAuth
   http://oauth.net/
   http://code.google.com/p/devdefined-tools/wiki/OAuth

8. XML
   http://es.wikipedia.org/wiki/Extensible_Markup_Language

8.3. Alguns enllaços d'interès a Internet

http://www.exist-db.org
eXist XML:DB.
http://www.rpbourret.com/xml/XMLDatabaseProds.htm
XML Database Products.
http://www.w3.org
W3 Consortium. Organisme encarregat d'assentar les bases de la web semàntica i els diversos llenguatges implicats.
http://www.w3.org/2001/sw
Secció del W3 Consortium encarregada de la part que correspon a la web semàntica.
http://www.w3.org/TR/rdf-sparql-query/
Secció del W3 Consortium encarregada del llenguatge de consulta SPARQL.
http://www.w3.org/XML/Query/
Secció del W3 Consortium encarregada del llenguatge de consulta XQuery.

http://www.slideshare.net/fulvio.corno/ontologies-introduction-designlanguages-and-tools
http://www.slideshare.net/snyderp/intro-to-the-semantic-web-peter-snydercsg339-northeastern-university-1697429
http://www.slideshare.net/LeeFeigenbaum/intro-to-the-semantic-weblandscape-2011

Jobs: https://www.linkedin.com/job/home
API Jobs: https://developer.linkedin.com/docs/DOC-1322
API Companies: https://developer.linkedin.com/documents/companylookup-api-and-fields
API LinkedIN: https://developer.linkedin.com/apis

http://www.hipertexto.info/documentos/ontologias.htm