



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat d'Informàtica de Barcelona



MENTOR INTELIGENT D'ITINERARIS FORMATIUS

MIQUEL UMBERT BOSCH

Director/a: KARINA GIBERT OLIVERAS (Departament d'Estadística i Investigació Operativa)

Titulació: Grau en Enginyeria Informàtica (Computació)

Memòria del projecte

Facultat d'Informàtica de Barcelona (FIB)

Universitat Politècnica de Catalunya (UPC) - BarcelonaTech

26/06/2023

Resum

Amb l'auge de la IA en els darrers anys i en una societat on tot està de cada cop més connectat i digitalitzat, ha augmentat la demanda de l'eficàcia i velocitat de tot tipus de programes. Aquest augment, ha significat un impuls tecnològic molt gran, fent que de cada cop els programes siguin més eficients i sofisticats. Per intentar comprendre el funcionament de les IA, s'ha decidit crear un programa que recomana itineraris formatius personalitzats a usuaris. Per realitzar aquest projecte, s'ha fet una compilació de dades de cursos oferts per diferents entitats. Després es va crear una ontologia de cada conjunt de dades perquè el nostre programa pugui realitzar la recomanació. Un cop tinguem l'ontologia i les dades dels cursos carregats al programa, només faltaria introduir dades d'usuari i el programa podrà realitzar una recomanació. Més endavant la idea és que el programa creàs les ontologies per ell mateix.

Agraïments

Vull agrair a totes les persones que han estat involucrades i m'han ajudat a l'hora de realitzar aquest projecte. Especialment a la meva directora del projecte Karina Gibert Oliveras per tot el temps i ajuda que m'ha donat. Aquest projecte no hauria estat possible sense que ella em guies pel camí adequat a l'hora de prendre decisions crucials que haurien canviat significativament el projecte. Gràcies a ella he tingut l'oportunitat d'aprendre conceptes únics. A més, també vull donar les gràcies al meu company Pablo Vega Gallego per ajudar-me a trobar un projecte que m'interessés com ha fet aquest.

Per una altra banda, voldria donar les gràcies a totes les persones de la xarxa pel seu treball en relació al camp de la intel·ligència artificial. Sense les seves aportacions, hauria estat bastant complicat aprendre noves tecnologies. Finalment vull agrair al centre de recerca Intelligent Data Science and Artificial Intelligence de la Universitat Politècnica de Catalunya per haver-me fet capaç de realitzar un projecte d'aquesta magnitud.

Índex de Continguts

1. Introducció.....	9
1.1 Context i abast.....	9
1.1.1 Context	9
1.1.2 Actors implicats.....	11
1.2 Justificació.....	12
1.3 Abast del projecte.....	13
1.3.1 Objectius.....	13
1.3.2 Requisits	13
1.3.3 Obstacles.....	14
1.4 Identificació de lleis i regulacions.....	14
2. Metodologia i rigor	15
2.1 Introducció	15
1.4.2 Eines.....	16
1.4.3 Validació	16
2.2 Disseny de l'eina.....	16
3. Disseny funcional i pla de treball	19
Descripció de les tasques.....	19
3.1 GP - Gestió de projectes.....	19
3.2 TP - Treball Previ.....	20
3.3 D - Disseny de l'eina	20
3.4 CRD - Creació de repositori de dades.....	20
3.5 VEO - Visualització i edició d'ontologies	20
3.6 CO - Creació de l'ontologia	20
3.7 U - Usuari.....	21
3.8 PREC - Procés de recomanació	22
3.9 IA - Interpretació i anàlisis resultats.....	22
3.10 DOC - Documentació	22
3.11 PR - Presentació.....	22
4 Creació de l'ontologia de referència	23
4.1 Introducció	23
4.2 Node.....	24
4.3 Graph	25
4.4 Importació ontologia	26
5. Repositori de dades	28
6. Visualització i edició ontologies.....	28
6.1 Introducció	29
6.2 Pantalla ontologia	31
6.3 Visualització.....	33
6.4 Creació ontologies	35
6.4.1 Nova ontologia.....	35
6.4.2 Importar ontologia	38
6.5 Edició d'ontologies.....	40

6.6 Funcionament intern	42
6.6.1 Creació de nodes	42
6.6.2 Visualització de l'ontologia	43
7. Usuari	45
7.1 Pantalla de dades	45
7.2 Pantalla usuari	53
7.2 Emmagatzematge de dades	63
7.2.1 Introducció	63
7.2.2 Item	64
7.2.3 Item Manager	66
7.3 Importació de dades	67
7.3.1 Introducció	67
7.3.2 Lectura fitxer	67
7.4 Connexió amb l'ontologia	69
7.5 Visualització dels ítems	70
7.5.1 Ocultar/Mostrar columnes	71
8. Recomanació	72
8.1 Introducció	72
8.1.1 Case based reasoning i K-Nearest Neighbors (KNN)	72
8.1.2 Distància Semàntica de Gibert (superconcept based distance)	73
8.2 Pre càlcul de distàncies a l'ontologia	74
8.3 Obtenció de dades de l'usuari	74
8.4 Recomanació de cursos	75
8.5 Casos d'ús i proves de concepte	76
8.5.1 Casos d'ús	78
9. Conclusió del projecte	80
9.1 Línies futures	81
A Planificació temporal	82
A.1 Planificació	82
A.2 Recursos	83
A.2.1 Recursos Humans	83
A.2.2 Recursos Hardware	83
A.2.3 Recursos de software	84
A.2.4 Recursos Generals	84
A.3 Planificació estimada i real	87
A.3.1 Gestió del risc	87
B. Pressupost i sostenibilitat	88
B.1 Identificació dels costos	88
B.1.1 Costos humans	88
B.1.2 Costos de software	89
B.1.3 Costos de hardware	89
B.1.4 Estimació costos del projecte	90
B.2 Control d'estimacions	93
B.3 Informe de sostenibilitat	94

B.3.1 Opinió personal	94
B.3.2 Aspecte ambiental	95
B.3.3 Aspecte social	96
B.3.4 Impacte econòmic	96
Bibliografia	97

Índex de Figures

Imatge 1: Logo DIH	12
Imatge 2: Exemple Trello	16
Imatge 3: Diagrama del programa	18
Imatge 4: Diagrama del programa	19
Imatge 5: Representació ontologia	25
Imatge 6: Exemple fitxer d'ontologies	28
Imatge 7: Exemple visualització d'ontologia	30
Imatge 8: Exemple ontologia en arbre	31
Imatge 9: Pantalla de visualització i edició d'ontologies	32
Imatge 10: Pantalla ontologia amb elements remarcats	33
Imatge 11: Marc representació de les ontologies	34
Imatge 12: Exemple d'ontologia simple	35
Imatge 13: 1r pas en la creació d'un node	37
Imatge 14: 2n pas en la creació d'un node	37
Imatge 15: Resultat del procediment de creació	38
Imatge 16: Exemple d'ontologia completa creada a partir de 0	39
Imatge 17: Botó importació ontologies	40
Imatge 18: Resultat d'importar ontologia	41
Imatge 19: Exemple de lectura d'ontologia	43
Imatge 20: Visualització de com pintam els nodes	45
Imatge 21: Pantalla de tractament de les dades	46
Imatge 22: Panell lateral d'ontologies	47
Imatge 23: Panell lateral amb ontologia creada	48
Imatge 24: Panell de visualització de la base de dades	49
Imatge 25: Panell de dades amb la base de dades importada	50
Imatge 26: Panell final de la pantalla de dades	51

Imatge 27: Panell final de dades amb elements identificats	51
Imatge 28: Pantalla per escollir l'arxiu de dades	52
Imatge 29: Finestra d'elecció de les columnes a visualitzar	53
Imatge 30: resultat d'aplicar el filtre	53
Imatge 31: Pantalla de l'usuari	55
Imatge 32: Menú superior de la pantalla de l'usuari..	56
Imatge 33: Finestra objectius	56
Imatge 34: Segona part de la pantalla de l'usuari.	58
Imatge 35: Finestra amb objectius exemple	59
Imatge 36: Cursos recomanats	59
Imatge 37: Finestra d'objectius amb 2 termes no identificables.	60
Imatge 38: Missatge d'error	60
Imatge 39: Finestra per guardar el PDF.	61
Imatge 40: Resultats de la creació del PDF	62
Imatge 41: PDF amb les dades de recomanació guardades..	62
Imatge 42: Diagrama UML classe abstracta.	66
Imatge 43: Representació de com veu el programa un fitxer .csv..	68
Imatge 44: Funció que retorna l'arbre de l'ontologia	70
Imatge 45: Gestió de visualització les columnes.	72
Imatge 46: Càlcul distància entre 2 tags.	74
Imatge 47: Fragment de la part de Data Science.	77
Imatge 48: Fragment de la part de Data Science.	78
Imatge 49: Cursos recomanats 1r cas	79
Imatge 50: Cursos recomanats 2n cas	80
Taula 1: Resum tasques projecte	86
Taula 2: Diagrama de Gantt	87
Taula 3: Salari anual de cada treballador del projecte	89
Taula 4: Costos hardware	90
Taula 5: Costos de les tasques	91
Taula 6: Costos generals del projecte	92

Taula 7: Pressupost final del projecte 92

1. Introducció

1.1 Context i abast

1.1.1 Context

Podem veure que la intel·ligència artificial és una eina relativament nova que ha crescut significativament en els darrers anys, fins i tot ha arribat la idea que aquest pugui suplantar llocs de treball de persones especialitzades, és una eina molt potent que es pot aprofitar de moltes maneres i en aquest projecte la volem posar en pràctica i veure que és capaç.

El projecte es desenvolupa en el context de [IDEAI-UPC](#)[1]: centre d'investigació en Ciència Intel·ligent de Dades i Intel·ligència Artificial de la UPC, amb les instal·lacions devora de la FIB (al Nexus II), Karina Gibert és la directora del centre i d'aquest TFG i alhora professora de la FIB.

En un context on la formació online agafa cada cop més volada, una de les línies de treball de la direcció del projecte és l'aplicació d'eines intel·ligents a la construcció del que hem anomenat "mentor intel·ligent" que en el fons són recomanadors basats en IA per itineraris formatius. Així, el projecte consisteix a dissenyar un mentor intel·ligent que utilitza una intel·ligència artificial per donar suport a la presa de decisions segons els criteris dels usuaris que la utilitzaran per fer consultes sobre quins cursos d'un cert repositori els hi convenen més, en funció de les seves preferències. Aquest haurà de fer una recomanació d'itineraris formatius segons la informació aportada per un usuari.

Aquesta línia de recerca s'està treballant a IDEAI-UPC des de diferents projectes:

- [Digital Innovation Hub](#)[2] (DIH4CAT): ecosistema regional d'innovació a Catalunya. El DIH4CAT té un mòdul transversal de formació que gestiona la formació oferta per tots els nodes del Hub en diferents cursos i que està coordinat per la UPC, en el que IDEAI té l'encàrrec de construir el mentor intel·ligent dels cursos del Hub.
- Professionalització de l'acceleració d'emprenedors amb eines d'IA: es tracta d'una col·laboració entre una empresa catalana que no podem revelar i IDEAI, dirigida per Karina Gibert i finançada en el marc de la convocatòria de Cupons Indústria 4.0 de testatge de tecnologies digitals avançades del propi Digital Innovation Hub. En aquest projecte es vol desenvolupar també un recomanador intel·ligent de formacions per emprenedors

El projecte s'ha construït amb l'aportació de bases de dades dels cursos que s'han obtingut d'una base de dades de [Kaggle](#)[3] i una vegada obtinguda la informació de l'usuari, es realitzaran models i tècniques de processament de dades i d'intel·ligència artificial per analitzar les dades i obtenir coneixement important per la presa de decisió a l'hora de recomanar el curs adient.

L'usuari participa indicant els seus objectius d'aprenentatge, per exemple: el tema principal que vol aprendre, les hores que pot dedicar, idioma preferent. Un cop obtinguda la informació, el mentor utilitzarà diverses tècniques de processament de dades per treure la solució que més criteris cobreixi.

Finalment IDEAI serà l'encarregat de dissenyar aquest mentor per als dos projectes esmentats i que sigui capaç de funcionar correctament i complir les expectatives amb diferents repositoris de cursos de diferents orígens i formats. La meua participació en la línia de recerca d'IDeAI és desenvolupar el cor d'un recomanador intel·ligent d'itineraris formatius flexible que permeti la fàcil adaptació a diferents corpus de cursos, sota la direcció de Karina Gibert.

Degut a tot això, aquest projecte pertany a la menció de computació del grau de informàtica. El projecte es durà a terme en les instal·lacions del centre IDEAI (Intelligent Data Science and Artificial Research Center) localitzat en l'edifici Nexus II, just devora de la FIB. Karina Gibert, directora d'IDeAI, estarà al corrent en tot moment de la situació del projecte.



Imatge 1: Logo del Digital Innovation Hub de Catalunya

1.1.2 Actors implicats

En el projecte hi participen diversos individus, cadascun amb unes funcions concretes:

- *Personal del mòdul de formació del i del node de IA del DIH4CAT:* Ens faciliten la informació dels cursos i de com necessiten que s'integri el mentor intel·ligent en la plataforma del DIH4cat.
- *Personal del projecte:* Aquí seríem nosaltres, l'autor i la directora del projecte, ens encarreguem de dissenyar i realitzar el projecte, provant diferents models i tècniques per obtenir bons resultats.
- *Els propis usuaris:* els usuaris que utilitzaran el resultat final, tindran un pes rellevant, ja que per a que funcioni, hauran de facilitar les dades necessàries per a que el mentor pugui elegir quin curs és més adient segons les seves necessitats, per tant seran els principals beneficiats del resultat del projecte.

1.2 Justificació

Com hem dit abans, la intel·ligència artificial cada cop té més pes en la societat, i les seves utilitats no s'aturen de créixer, és cert que encara hi ha limitacions que afecten el seu ús. Però no és cap sorpresa que de cada vegada aquesta tindrà un pes més important en el nostre dia a dia, per tant la seva incorporació directa a la societat és cada dia més imminent, cada dia surten nous articles que debaten sobre l'ètica i l'ús d'aquesta tecnologia, però és un procés que no es pot evitar.

Empreses i universitats de tot el món estan fent una recerca i estudis massius de com aprofitar els beneficis que pot aportar la IA en el nostre dia a dia, empreses destacades com [Microsoft](#)[4], [Apple](#)[5], [Google](#)[6], [Facebook](#)[7] han fet grans inversions directament en la IA o comprant empreses ja més expertes en aquest àmbit com diuen els articles anteriors. Per tant podem veure que hi ha bastant de potencial.

També hi ha la contrapart, gran part de la població certament queda asombrada de l'ús d'aquesta tecnologia, però alhora molta gent demostra temor. Hi ha infinitat d'articles que debaten si la IA és un benefici o si suposa una amenaça. Aquí tenim dos exemples [\[8\]](#) [\[9\]](#), el primer parla de l'amenaça de la IA davant l'educació, on cita que aquesta pot resoldre preguntes i operacions complexes, com a contrapart, el segon cita els principals avanços en la medicina gràcies a la IA.

Si aquestes es van implantant poc a poc i la població veu resultats, és possible que aquest temor desaparegui i de cada vegada veiem més articles que destaquen les seves forteses.

Encara que la IA és una eina relativament nova, degut al seu gran impacte i possibilitats, hi ha una gran quantitat d'informació útil que podem fer servir en el projecte, des de les assignatures impartides en la FIB a complexos articles que la donen a conèixer. Avui en dia ja tenim diverses [aplicacions web](#)[10] que funcionen utilitzant IA, per tant podem agafar referència d'ells també. Però en el nostre cas de fer un recomanador d'itineraris que utilitzi intel·ligència artificial, no ens hem trobat cap plataforma que faci ús de la IA en la seva web, si bé és cert que hi ha certs recomanadors d'itineraris, com per exemple [udeka](#)[11], no podem assegurar que utilitzen intel·ligència artificial per realitzar les conclusions, sembla que aquestes webs utilitzen només certs paràmetres de filtració, però realment no fan una recomanació personalitzada. Per tant podem arribar a la conclusió que la nostra solució ja existeix però utilitzant una tecnologia diferent, ja que el seu cas no es fa un ús de la IA, en el nostre sí que volem que se'n faci ús, perquè ens donarà resultats més encertats i vertaders.

Degut a tot això, podem dir que aquest projecte pertany a la menció de computació del grau de informàtica, ja que utilitzarem tècniques i eines, com la IA, processament de dades... que es donen bastant en aquesta menció.

El projecte es durà a terme en les instal·lacions del centre IDEAI (Intelligent Data Science and Artificial Research Center) localitzat en l'edifici Nexus II, just devora de la FIB.

Karina Gibert, directora d'IDeAI, estarà al corrent en tot moment de la situació del projecte.

1.3 Abast del projecte

1.3.1 Objectius

Com s'ha dit al principi, la intel·ligència artificial està progressant a un gran ritme, per això l'integració d'aquesta a la societat es fundamental. Aquest projecte té l'objectiu de mostrar un dels múltiples beneficis de la IA. Per tant, ens sorgeixen els següents objectius:

- Crear un mentor intel·ligent capaç de recomanar itineraris personalitzats segons els criteris d'un usuari. El objectius de formació expressats per d'un usuari, que sigui capaç d'anar més enllà de les clàssiques seleccions de paraules clau. En el projecte es tindrà en compte l'ús d'ontologies de referència que permetin tractar la semàntica de les paraules i entendre per exemple que "machine learning" està més relacionat amb "intel·ligència artificial" que amb "blockchain" i que "ciberseguretat" és una competència clau en "tecnologies de núvol" per exemple.
- Realitzar mètodes de processament de dades i d'intel·ligència artificial per obtenir resultats esperats
- Aprendre models basats en dades. En altres assignatures hem realitzat alguns exemples semblants a aquest.
- Adquirir coneixement en el món de la IA, ja que aquest pot ser essencial en el futur si la tecnologia va evolucionant com està fent ara.

1.3.2 Requisits

Per tenir un bon projecte, s'han de complir els requeriments que considerem necessaris:

- Tenir una bona base de dades de cursos, amb tots els seus camps i columnes corresponents, sense que ens faltin dades.
- Bona elecció de les dades que ens proporcionarà l'usuari, les dades que aquest ens donarà, seran essencials a l'hora de crear models i elegir un curs adient, per això no hi pot haver errors de format ni falta d'informació essencial per part d'aquest
- Elegir bons models per a les dades proporcionades, sense un bon model que s'ajusti a les dades, serà impossible tenir un resultat convincent.
- Crear una bona documentació del projecte, no hi haurà un bon projecte sense una bona documentació que l'acompanyi. Si només entén el treball el mateix autor, queda obsolet i en desús.
- Seguir bones pràctiques de programació, saber quines classes s'han de crear, que quedi el codi ben estructurat, que sigui simple i fàcil d'entendre, per si en un futur s'han de fer modificacions, que qualsevol pugui entendre i modificar el còdi fàcilment.

- Anar al dia amb la planificació, un projecte d'aquestes dimensions sense una bona planificació i estructuració de tasques, està condemnat al fracàs total, per això es realitzarà un planning de les tasques que es convenient seguir

1.3.3 Obstacles

Durant la realització del projecte poden sortir una sèrie d'obstacles que s'han de tenir en compte, els més destacats que em tret són:

- Inexperiència de l'autor del projecte. L'autor ha fet model i treballs simples relacionats amb la intel·ligència artificial, però mai un a aquest nivell.
- Obtenció de les dades, per poder realitzar el treball són necessaris els cursos que s'impartiran.
- Mala informació de l'usuari, es possible que algun usuari a l'hora de facilitar les seves dades, les doni de forma incorrecte, cosa que pot provocar fallides no contemplades.
- Quantitat de temps, l'autor del treball també té altres assignatures que en general no suposen un gran perill, però hi pot haver moments on la càrrega de treball sigui més gran de l'esperat i no es pugui dedicar el temps necessari a la realització del treball.

1.4 Identificació de lleis i regulacions

En aquest projecte, a la llarga haurem de tractar dades de diferents entitats i també de persones que vulguin una recomanació, així que necessitarem dades personals seves, per lo tant el projecte està regulat per la [lleï de protecció de dades de la UE](#)[12]. Com que no compartirem les dades de forma insegura amb cap entitat més, podem dir que el nostre projecte compleix amb aquesta llei.

Per una altra banda, en dur a terme un projecte relacionat amb IA, hi ha certes lleis i etiquetes que la [Comissió Europea](#)[13] regula, en el nostre cas també es compleix amb aquesta llei/regulació. El passat 14 de juny el Parlament Europeu va aprovar el Reglament Europeu de la IA que es trasllada ara al Consell d'Europa.

Per tant, sota lleis i regulacions, el nostre projecte compleix amb tot.

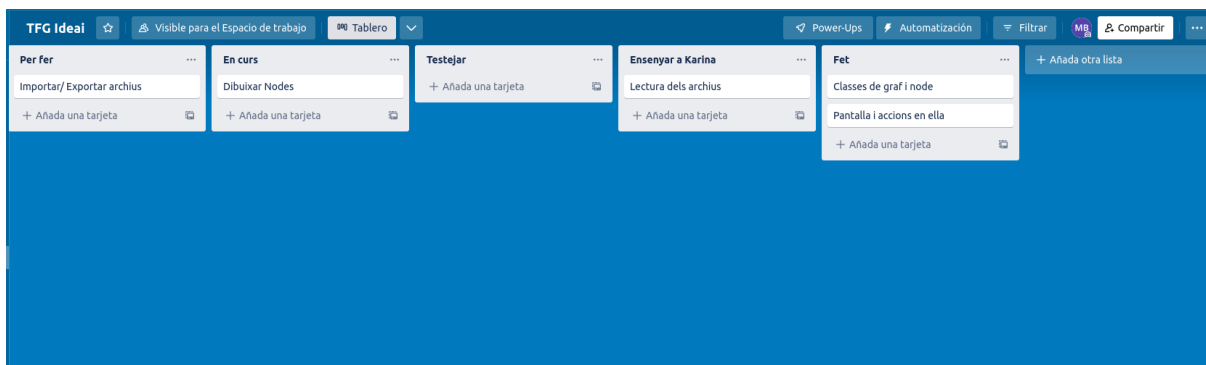
2. Metodologia i rigor

2.1 Introducció

En aquest projecte utilitzarem una metodologia anomenada [Kanban](#)[14], que tracta de llistes amb targetes, on aquestes són les tasques a realitzar, i les llistes són l'estat actual de la tarea. En el nostre cas, tenim 5 possibles llistes:

- Per fer: les tasques que no s'han fet i estan esperant a ser realitzades.
- En curs: les tasques que s'estan realitzant en aquest moment.
- Testejar: tasques acabades però que no s'ha comprovat correctament el seu funcionament.
- Ensenyar al director: tasques acabades i que funcionen correctament però que han de ser aprovades pel director del projecte per ser considerades fetes.
- Fet: tares que han estat aprovades pel director del projecte.

Cercant per Internet, hi ha diverses webs que ofereixen un model com aquest, però en aquest cas, em elegit la pàgina [Trello](#)[15], que està orientada a ajudar als equips a organitzar i realitzar un seguiment de la seva feina. Ens hem decantat sobretot per la bona experiència anterior amb aquesta pàgina. En la següent figura, mostra un exemple de trello.



Imatge 2: **Exemple de Trello** amb el nostre treball.

Cal destacar que segurament hi haurà tasques que no passaran obligatòriament per totes les fases, però seran tasques excepcionals. Com es pot veure, la nostra metodologia és fàcil i intuïtiva, on es pot dur un bon seguiment del projecte i de les tasques que el componen.

1.4.2 Eines

Durant el projecte utilitzarem diverses eines, aquí ens referim sobretot a llenguatges de programació, entorns de treball i control de les versions del projecte.

Llenguatges de programació

De moment el principal i únic llenguatge del projecte és [java](#)[16], però en un futur, no es descarta la idea de utilitzar altres llenguatges conjuntament, sobretot llenguatges com [R](#)[17] o [Python](#)[18], ja que aquests tenen moltíssimes llibreries per fer els models de processament de dades i d'intel·ligència artificial.

Entorns de treball

L'entorn de treball de moment es [IntelliJ Idea](#)[19], un IDE desenvolupat per [JetBrains](#)[20]. Ja que és el millor IDE per realitzar projectes amb Java, més endavant si s'arriben a aplicar nous llenguatges, segurament s'afegiran nous entorns de treball al projecte.

Control de versions

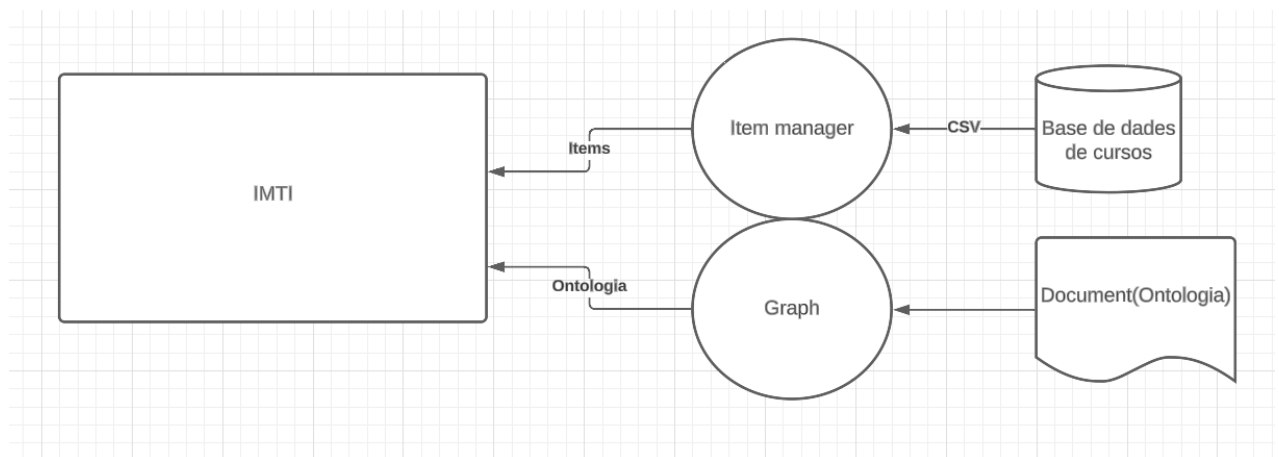
Finalment, pel control de versions del projecte, hem escollit [Git](#)[21], per la seva fiabilitat i el seu gran ús en el món de la informàtica. De moment només tenim una rama pel projecte, però a mesura que surtin tasques més complexes, la idea es crear una rama per cadascuna d'aquestes tasques més complexes.

1.4.3 Validació

Per la validació del projecte, es pot fer de forma immediata mirant el progrés de les tasques a Trello. Però degut a que el treball es realitza a les instal·lacions de IDEAI on la directora també treballa, aquesta podrà consultar l'estat del projecte en qualsevol moment sense haver d'acordar una reunió.

2.2 Disseny de l'eina

En aquest apartat ensenyarem les diferents parts del programa d'una forma més simple i comprensible:



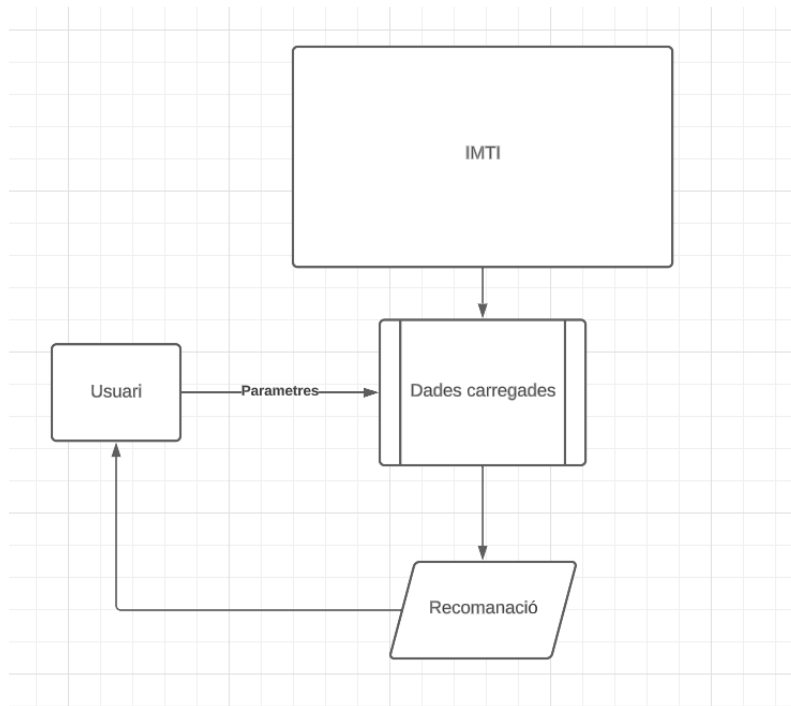
Imatge 3: **Diagrama del programa**, carregament de dades

En la imatge superior podem apreciar els 3 grans components del programa, per començar tenim el recomanador (IMTI), que rebrà les dades de l'ítem manager i del graph per tenir les dades carregades al sistema.

L'ítem manager ens proporcionarà una base de dades dels ítems que vulguem recomanar, en el nostre cas seran els cursos, per a que l'ítem manager pugui generar aquestes dades, primer se li ha d'introduir la base de dades que vulguem en forma de csv. Aquest llegirà el document i generarà els ítems, d'aquesta forma el programa principal tindrà accés a les dades dels ítems que es voldran recomanar.

Per una altra banda tenim el mòdul de graph, aquest és l'encarregat de proporcionar l'ontologia al programa, que ajudarà a fer la recomanació que l'usuari desitgi. Aquest per generar l'ontologia, haurà de llegir un document de text per a construir l'ontologia, o bé es podrà introduir manualment de forma gràfica i fer l'ontologia des de 0 o modificar una ja existent.

Un cop s'han carregat aquests paràmetres, que venen a ser la base de cursos dels quals es vol realitzar la recomanació i l'ontologia que guiarà al programa a l'hora de realitzar la recomanació.



Imatge 4: **Diagrama del programa**, obtenció paràmetres usuari i recomanació

Aquest seria el pròxim pas en el procés de recomanació.

Com podem veure el programa ja té les dades necessàries per a funcionar carregades. Els únics paràmetres que falten per a realitzar una recomanació són els objectius que l'usuari vol assolir. Una vegada els ha facilitat, el programa generarà una recomanació, i l'usuari podrà escollir un dels cursos recomanats.

3. Disseny funcional i pla de treball

Descripció de les tasques

En aquest apartat descriurem les tasques que és duran a la llarga dins del projecte. Cal dir que les hores que durarà cada tasca han estat només una estimació. Si una tasca dura o es complica més de l'esperat, l'autor haurà d'invertir més hores de feina diàries de les que toca.

3.1 GP - Gestió de projectes

-GP.1 Contextualització i abast

Com sabem el treball de fi de grau és bastant complexe i hem de invertir algo de temps per fixar uns límits i característiques que aquest ha d'assolir per complir el nostre objectiu. D'aquesta forma tindrem els objectius clars i sabrem el que hem de fer en tot moment.

La guia de GEP indica que la durada és de unes 25 hores.

-GP.2 Planificació temporal

Es realitzarà una planificació total del temps, recursos i tasques que farem al llarg del projecte. Aquesta part segons la guia té una durada de 9 hores.

-GP.3 Pressupost i sostenibilitat

Es realitzarà un pressupost total del projecte del que obtindrem el cost del projecte, i es detallarà els costos que ha tingut cada part del projecte. En la sostenibilitat es veurà el possible impacte econòmic, ambiental i social que pot tenir el nostre projecte de cada a la societat actual.

L'estimació segons la guia és de 9 hores.

-GP.4 Document final

És la unió dels tres documents anteriors, tenint en compte els comentaris que el tutor de GEP hagi fet. Estimació de 19 hores seguint la guia.

-GP.5 Reunions

La comunicació entre l'autor i el director del projecte és molt important, per poder comprar i analitzar els resultats aquests han de fer reunions, del contrari seria molt complicat dur el seguiment del projecte. Aquests tenen l'avantatge que estan al mateix edifici, saltant-se la necessitat d'organitzar reunions de forma precisa. Temps estimat: 15 hores.

3.2 TP - Treball Previ

Es farà un recerca prèvia per veure models i/o exemples actuals que puguin ajudar a desenvolupar el projecte de forma efectiva. Es una tasca simple pero que cal invertir un nombre considerable d'hores per poder cobrir tots els aspectes possibles. Temps estimat: 18 hores.

3.3 D - Disseny de l'eina

Farem un estudi de com volem que estigui organitzada l'eina que vol crear, les pantalles, les classes necessàries, la seva comunicació, i lo que pugui sortir. Temps estimat: 5 hores.

3.4 CRD - Creació de repositori de dades

Per realitzar les proves i comprovar el funcionament del programa, necessitarem trobar una base de dades de cursos útil i adaptar-la per a què la poguem utilitzar durant les proves. Temps estimat: 2 hores.

3.5 VEO - Visualització i edició d'ontologies

-VEO.1 Organització i documentació de la tasca

Degut a que haurem de crear una IA que recomani itineraris, aquesta es basarà en una ontologia, per poder facilitar la visualització d'aquesta, crearem un programa que permeti visualitzar ontologies i editar-les en forma de nodes connectats entre si. Un cop ja sabem que hem de fer, ara cal cercar la forma d'implementar-ho. Temps estimat: 20 hores.

-VEO.2 Programa i finestra de visualització

Usant el software adient, crearem l'estructura del programa i crearem una finestra per visualitzar els nodes i les seves relacions. Temps estimat: 5 hores.

-VEO.3 Funcions finestra

Crearem funcions per a que la finestra sigui més eficient: crear i eliminar nodes de forma dinàmica, establir relacions entre nodes de forma dinàmica, importar i exportar ontologies en arxius de text, on hi haurà el nodes i les seves relacions. Temps estimat: 50 hores.

3.6 CO - Creació de l'ontologia

-CO.1 Organització i documentació de la tasca

Degut a la complexitat de la tasca, caldrà informar-se bé de com afrontar-la i barrejar diferents opcions per acabar escollint la que creiem més òptima. Temps estimat: 20 hores.

-CO.2 Creació de les classes de l'ontologia

Un cop ja sabem com afrontar la tasca, procedirem a realitzar-la de la forma més convenient, a més de cercar una forma de relacionar les dades que ens donaran dels cursos amb una implementació òptima de codi. A més que s'hauran de crear les classes adients per guardar la informació, ja sigui dels nodes de l'ontologia o de les dades en si. Temps estimat: 30 hores.

-CO.3 Models

Una vegada ja tinguem les classes i les dades necessàries, crearem models de processament de dades per realitzar la recomanació més adient de l'usuari. Aquesta tasca pot resultar bastant complexa, fins al punt que s'hagin de crear noves tasques derivades d'aquesta. Temps estimat: 30 hores.

3.7 U - Usuari

-U.1 Identificació de vistes i tipus d'usuari

Dins del programa, de moment hem considerat l'existència de dos tipus d'usuari:

- El manager, que s'encarregarà de la gestió de la ontologia i dels repositoris de cursos disponibles, que podrà modificar tant la ontologia com els cursos.
- Usuari comercial: serà l'usuari que vol realitzar una recomanació, per tant aquest només haurà d'especificar els seus objectius i el nombre de cursos que vol que se li recomanin

Temps estimat: 2 hores.

-U.2 Creació de classes

Necessitarem una classe que correspongui a l'usuari que està utilitzant el nostre programa i haurem de recollir les dades que aquest ens facilitarà per poder realitzar la recomanació que més s'ajusti a les seves necessitats. Temps estimat: 10 hores.

-U.3 Guia visualització de l'usuari

El projecte quedara molt més detallat si incorporam una interfície gràfica que comuniqui l'usuari directament amb el programa principal, per això s'hauran de crear pantalles, on unes recolliran la informació que donarà l'usuari i altres que mostraran els resultats de la predicció. Temps estimat: 38 hores.

3.8 PREC - Procés de recomanació

Una vegada tinguem tot montat, decidirem la millor estratègia per a la recomanació de cursos, ja sigui crear i fer les classes encarregades de fer la recomanació, quines dades necessitarà, com les obtindrà, com les processarà, etc. Temps estimat: 30 hores.

3.9 IA - Interpretació i anàlisi resultats

-IA.1 Proves de concepte

Es faran diverses proves amb exemples de possibles casos reals per obtenir resultats i validar el correcte funcionament del programa. Si els resultats són positius, farem un anàlisi i comparació entre diferents jocs de proves, cercant falles o casos concrets. Temps estimat: 10 hores.

-IA.2 Interpretació resultats

Una vegada tenim els resultats, validarem la seva correctesa i realitzarem una conclusió del treball fet fins al moment, indicant les conclusions i futures millores que es podran fer. Temps estimat: 10 hores.

3.10 DOC - Documentació

-DOC.1 Recollida d'informació

Durant el curs, anirem recollint informació important per poder incorporar-la al document final del projecte. Aquesta tasca durara durant tot el temps que estiguem fent el projecte. Temps estimat: 25 hores.

-DOC.2 Document final

Fent ús de la informació recollida i extreta dels anàlisis de les proves, realitzarem el document final del nostre projecte, Temps estimat: 50 hores.

3.11 PR - Presentació

Quan tinguem tot el projecte finalitzar, el programa principal i el document final, ens prepararem per a la defensa de la presentació del treball. Temps estimat: 15 hores.

4 Creació de l'ontologia de referència

En aquest apartat explicarem el procés que va suposar la creació del nostre model d'ontologia del nostre projecte, de formes que siguin fàcils d'interpretar, de visualitzar i de tractar.

Aquí explicarem les decisions que vam prendre per saber quins atributs otorgar a l'ontologia, les classes necessàries, les funcions de cada classe, etc.

4.1 Introducció

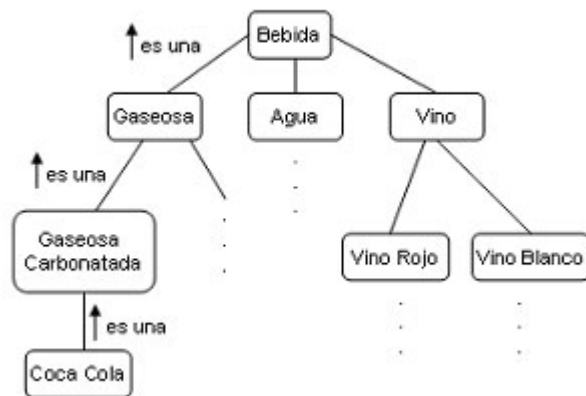
Per començar, hem d'entendre què són i de quina forma és representen les ontologies.

[La definició d'ontologia](#) [22] diu que és una representació formal i explícita del conceptes, bàsicament una estructura organitzada d'informació que conté el significat i les relacions entre els diferents elements d'un camp o àrea d'estudi. Permeten modelar el significat i la interpretació de dades, permetent una comprensió més profunda i precisa de la informació.

Les ontologies són utilitzades en diferents disciplines, com la intel·ligència artificial, la ciència de la informació, informàtica, web semàntica entre altres. I en el nostre cas, hem adaptat un model d'ontologia simple perquè sigui bastant comprensible, basant en una sèrie d'elements jerarquitats que van des de lo més general fins a lo més específic.

Per la seva comprensió, vam optar per fer un model basat en arbres, d'aquesta forma els conceptes estan organitzats en diferents nivells segons una jerarquia, l'arrel de l'arbre representarà el concepte més general del nostre domini, mentre que els nodes que surtin d'aquest s'especialitzaran en conceptes més específics a mesura que es va baixant de nivell.

Aquesta model permet una organització estructura jeràrquica del coneixement, lo que facilita la seva navegació i comprensió.



Imatge 5: **Representació d'una ontologia** seguint un model d'arbre

4.2 Node

La classe Node, serveix per representar els components que formen la nostra ontologia. En aquest apartat explicarem un poc els diferents atributs i utilitats que té la classe Node.

Aquesta classe és bastant simple i intuïtiva, això són els diferents atributs que té:

- Nom -> el nom amb que es representarà el component de forma visual
- Identificador -> identificador creat per nosaltres per una millor gestió a l'hora de treballar amb diferents nodes
- Node pare -> una referència al node que fa de paper de "pare", cal dir que els nodes només poden tenir un pare. Útil per saber les relacions.
- Conjunt de nodes fills -> un conjunt de referències als diferents nodes que són els "fills" d'aquest, ens serveix per guardar les relacions entre els diferents nodes. A conseqüència d'això els fills tindran aquest node com a "pare".
- Conjunt de noms de tots els ancestres -> definim com a ancestres els "pares" del nostre node pare, així fins a arribar al node arrel, que tots tindran com a ancestre. Aquesta propietat ens servirà per a més endavant.

Apart de les propietats, tenim les funcions bàsiques de getters i setters de la nostra classe.

Altres aspectes a destacar es que si el node és eliminat, aquest s'encarrega que els seus fills l'eliminini com a pare i que el seu pare l'elimini de fill. D'aquesta forma mantenim una estructura rígida i actualitzada.

A aquesta classe, no hem vist necessari agregar més atributs o funcions, certament l'atribut dels ancestres no és essencial per mantenir una ontologia, nosaltres l'hem creat ja que ens proporciona informació útil per realitzar la recomanació, per tant la classe node està bastant completa i unificada.

4.3 Graph

La classe Graph, serveix com un gestor de nodes i per tractar les seves relacions, apart de proporcionar informació sobre els nodes i sobre el conjunt en sí.

Aquesta classe guardarà tots els nodes creats dins l'ontologia, i s'encarregarà de proporcionar la informació necessària en tot moment que faci referència als nodes i les seves relacions.

Es comunica amb altres gestors per intercanviar informació i mantenir el conjunt de nodes actualitzat. Compleix les funcions bàsiques de proporcionar la informació que li sol·liciten i actualitzar els atributs d'un node si aquest ha estat modificat per algú.

Cal destacar que utilitza el [patró de disseny singleton](#) [23], és a dir, només pot ser instanciada un sol cop i proporciona un accés global a la instància de la classe.

Hem utilitzat aquest patró, ja que considerem que amb un sol gestor de nodes és suficient pel nostre projecte, així ens assegurem que no es creen més instàncies d'aquesta classe i ens evitam possibles errors.

Aquesta classe té més llibertat a l'hora de crear atributs i funcions, ja que al tractar un conjunt de nodes, es pot treure molta informació a partir d'aquests.

Nosaltres hem creat els atributs que hem considerat necessaris i les funcions que hem necessitat durant el projecte:

- HashMap nodes -> un hashmap que té per clau el nom del node i per valor la pròpia referència del node.

- HashMap nodesId -> un altra hashmap, l'única diferència és que la clau en aquest cas és l'identificador que nosaltres li donam al node.
- Graph singleton -> una referència a la pròpia classe graph, necessària per utilitzar el patró de disseny singleton.
- nodeId -> un enter iniciat a 0, cada cop que es crea un node a la nostra ontologia, se li assigna nodeId com a identificador, i després s'augmenta el valor d'aquest.

Aquestes serien les propietats de la nostra classe, com es pot veure no hi ha moltes propietats. Òbviament es poden crear moltes més propietats en lo que respecte a un graf de nodes, en el nostre cas no ha estat necessari definir més estructures dins la nostra classe.

L'ús de dos HashMaps pareix ineficient, ja que els 2 referencien al mateix objecte però amb diferent clau, la raó d'utilitzar els dos s'explica en el següent punt.

4.4 Importació ontologia

Durant el projecte vam decidir tenir la possibilitat de crear ontologies de forma ràpida i efectiva, per això s'ha creat una forma d'introduir dades de eficientment al nostre programa.

Es tracta de un simple arxiu de text al que donem una extensió concreta “.ont”, on se definiran els nodes i les seves relacions.

L'aspecte de l'ontologia és senzill i respon a un criteri d'expressions regulars.

Node arrel: fills,*

Fills: nodes fills,*

```

1 Pare: Informatica, Industria, Materials, innovacio;
2 Informatica: Ciberseguretat, Big Data, intel·ligencia artificial;
3 Industria: Organitzacio, Produccio, Industria 4.0;
4 Organitzacio: Gestio i habilitats, Organització industrial;
5 Produccio: disseny, fabricació additiva, manteniment industrial, mecanització;
6 Materials: omiques, plastics i compostes, nutricio i salut, motlles i matrius;
7 innovacio: transformacio digital;
8 intel·ligencia artificial: etica de la IA;
9 Industria 4.0: manteniment predictiu, bessons digitals;

```

Imatge 6: Exemple de fitxer de text d'ontologies.

En la imatge superior, tenim un model d'una ontologia en format de fitxer de text.

Podem veure la seva composició, al començament de cada fila, tenim el nom del node pare que volem, un cop acaba el nom del pare, es posa el signe de dos punts (:) per senyalitzar el final.

Seguit dels (:) comencem a enumerar el nom dels fills del respectiu node, separats per comes (,) entre ells, un cop hem acabat de numerar els fills, es finalitza la línia amb un punt i coma (;)

Cal mencionar que els nodes que siguin exclusivament fills, no importa crear una línia aposta per ells. El programa quan llegeix els fills, si aquests no tenen un node creat, el programa el crea per ells, per tant ens estalviem escriure línies on només hi hauria el nom del node a crear, sense cap fill.

Aquí entra en joc tenir els dos hashmaps anteriorment mencionats, a l'hora de llegir l'arxiu ".ont", tenim com a referència els noms dels nodes, no el seu identificador, al no saber quin identificador li posa a cada node, vam decidir tenir els dos hashmaps i utilitzar-los eficientment. Cal dir, que aquesta és l'única vegada en tot el programa que referenciarem els nodes pel seu nom, en tots els altres casos, ja sigui de petició de dades, modificació, eliminació, etc; utilitzarem l'identificador del node en lloc del seu nom.

5. Repositori de dades

Per a la realització d'aquest projecte, varem decidir cercar a la pàgina web de Kaggle una base de cursos.

Després de cercar i barrejar diferents opcions vam trobar aquesta base de cursos de [Udemy](#)[24], una plataforma d'aprenentatge en línia.

Aquesta base de dades la vam adaptar per a que funcionas en el nostre entorn, només va ser necessari afegir la columna de "Tags" i rellena-la amb nodes de l'ontologia perquè el programa fos capaç de realitzar la recomanació.

La base de dades no amaga cap misteri, té les columnes que són d'esperar:

- Preu del curs
- Nombre de participants
- Url del curs per a més informació
- Ressenyes fetes del curs
- Duració del curs
- Nivell del curs

S'ha de dir que el nostre programa llegeix les columnes de forma dinàmica, per tant ara mateix ens serveix qualsevol arxiu CSV que tingui una columna anomenada Tags i que aquests estiguin inclosos dins la ontologia que s'utilitzarà. Per tant el programa és fàcilment integrable a diferents ambients i a casi tot tipus de dades.

6. Visualització i edició ontologies

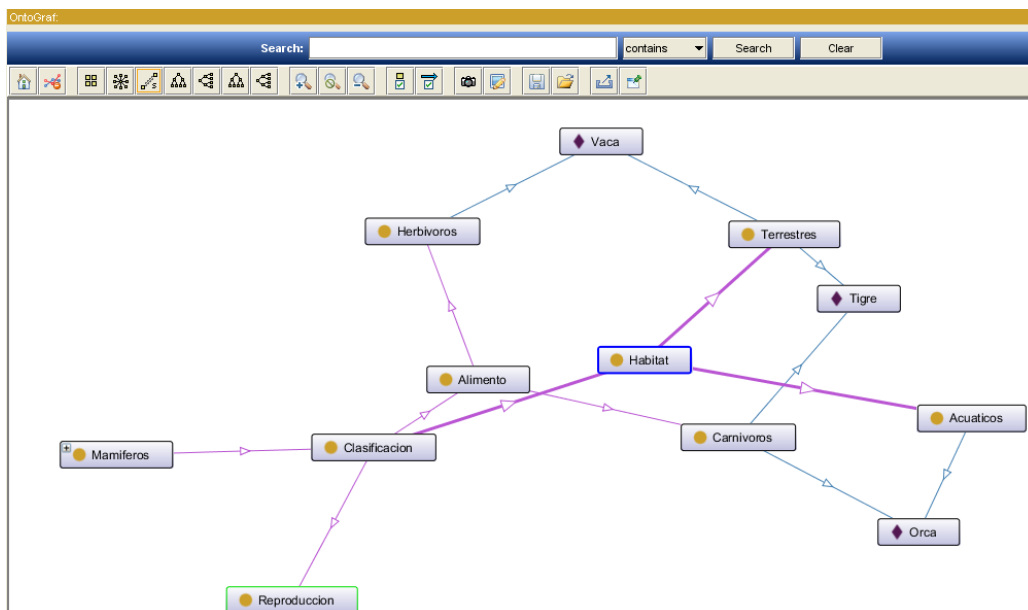
En aquest apartat explicarem el procés que va suposar visualitzar i editar les ontologies que es poden introduir al nostre projecte. I per tant que aquestes siguin visibles i entendibles per les persones no especialitzades en el camp i que aquestes mateixes les puguin editar de la forma que ells vulguin.

Un cop ja vam saber com gestionaríem la nostra ontologia, el següent pas va ser decidir com es visualitzaria, els elements que s'havien de mostrar, les coses que es poden modificar...

Veient que vam optar per seguir un model d'estructura d'arbre, format per nodes amb relacions entre ells, i que tot el conjunt de nodes i relacions formava un graf, una estructura molt vista durant el grau, van optar per la típica representació d'un graf en forma d'arbre: partint d'un node arrel i que a partir d'ell surten tots els altres nodes, cadascun especialitzat en un àmbit del domini.

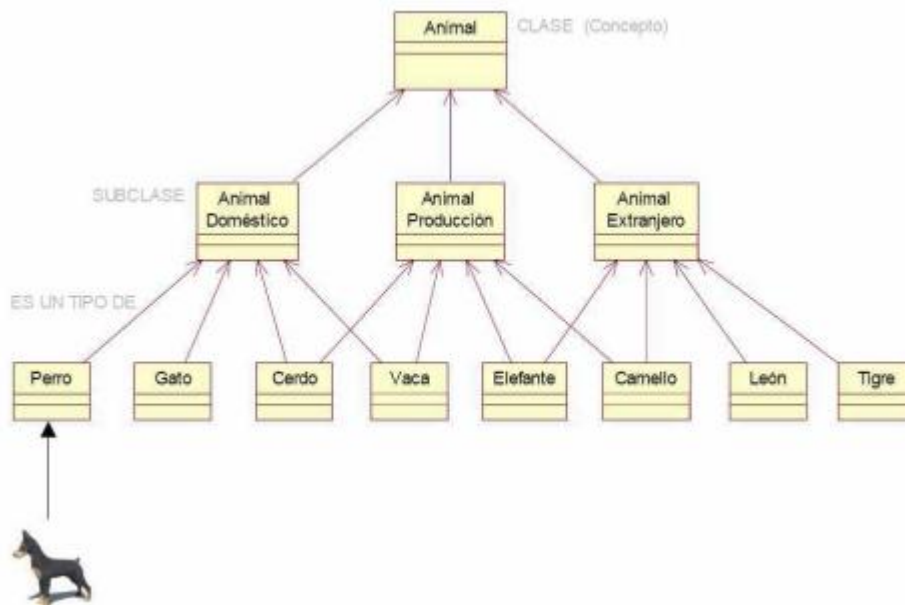
6.1 Introducció

Quant a la visualització de les ontologies, hi ha diferents mètodes i models a seguir per la seva visualització, la forma més simple és mitjançant una interfície gràfica per navegar dins l'ontologia i comprendre millor les relacions, cosa que facilita la identificació i interpretació de la informació. Els casos més comuns són nodes dispersos amb relacions, sense una jerarquia.



Imatge 7: Exemple de visualització d'ontologia.

Com podem veure a la imatge d'adalt, aquesta metodologia pot dur a confusions i incomprensió de les dades de l'ontologia, per això com hem mencionat abans, es va decidir seguir una estructura en forma d'arbre, on el node arrel està a la part més superior i d'ell deriven tots els subnodes que defineixen els conceptes més generals del nostre domini.

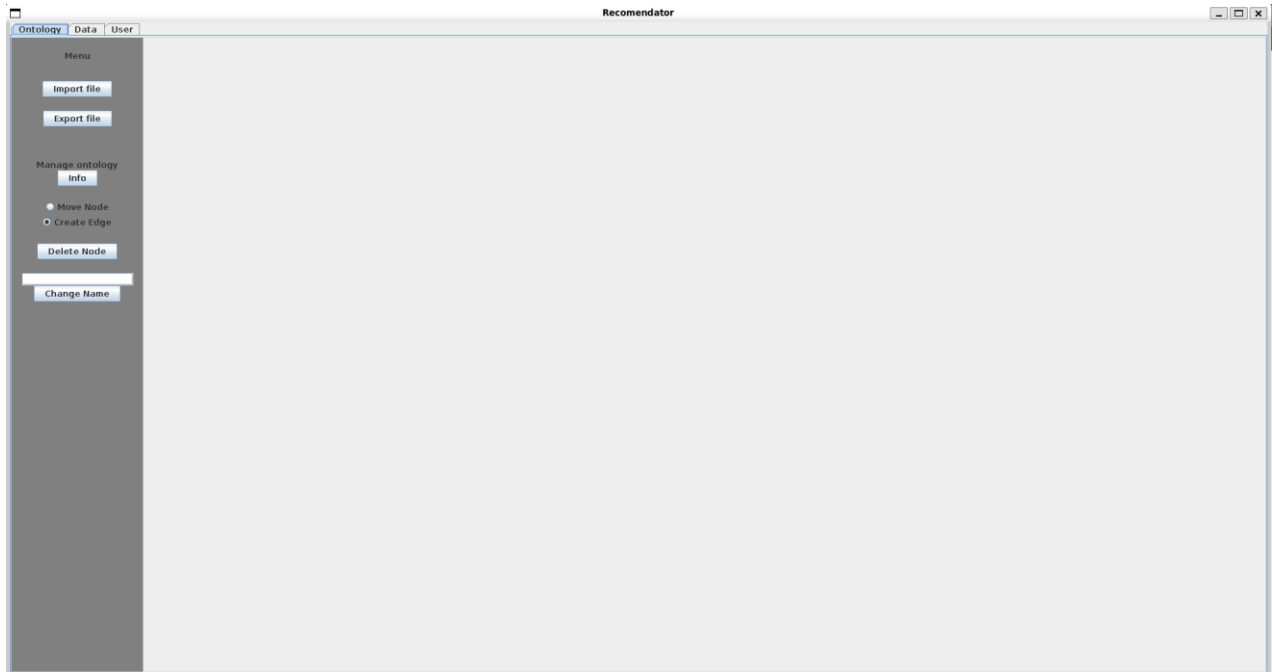


Imatge 8: Exemple d'ontologia en forma d'arbre

Considerem que pel nostre projecte aquesta és una forma més òptima de representar una ontologia, a més que el fet que hi hagi un node arrel que el comparteix tota l'ontologia, facilita el tractament de dades que d'urem a terme més endavant.

6.2 Pantalla ontologia

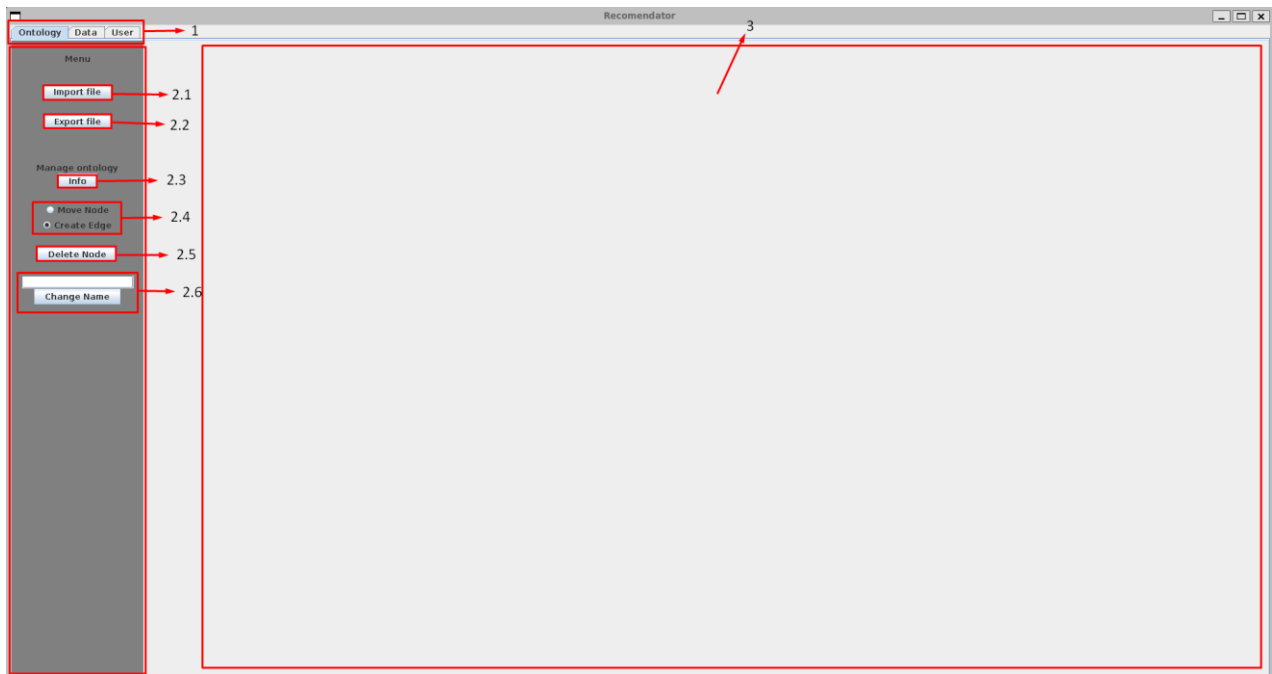
Aquí explicarem els diferents elements que componen la nostra pantalla de representació i edició d'ontologies, a més de la funcionalitat que tenen:



Imatge 9: **Pantalla de visualització i edició d'ontologies.**

En la imatge superior, podem veure com seria la nostra pantalla de visualització i edició d'ontologies. S'ha cercat que aquesta pantalla tenguí un estil minimalista, amb no molta càrrega d'elements visuals i colors freds per no carregar la vista de l'usuari que estigui utilitzant el programa.

A continuació tenim una imatge remarcant els diferents elements que hi ha a la pantalla i els seus subelements si es que en tenen, s'explicarà que és cada element i les funcions que aquest té dins del programa:



Imatge 10: **Pantalla ontologies amb elements remarcats.**

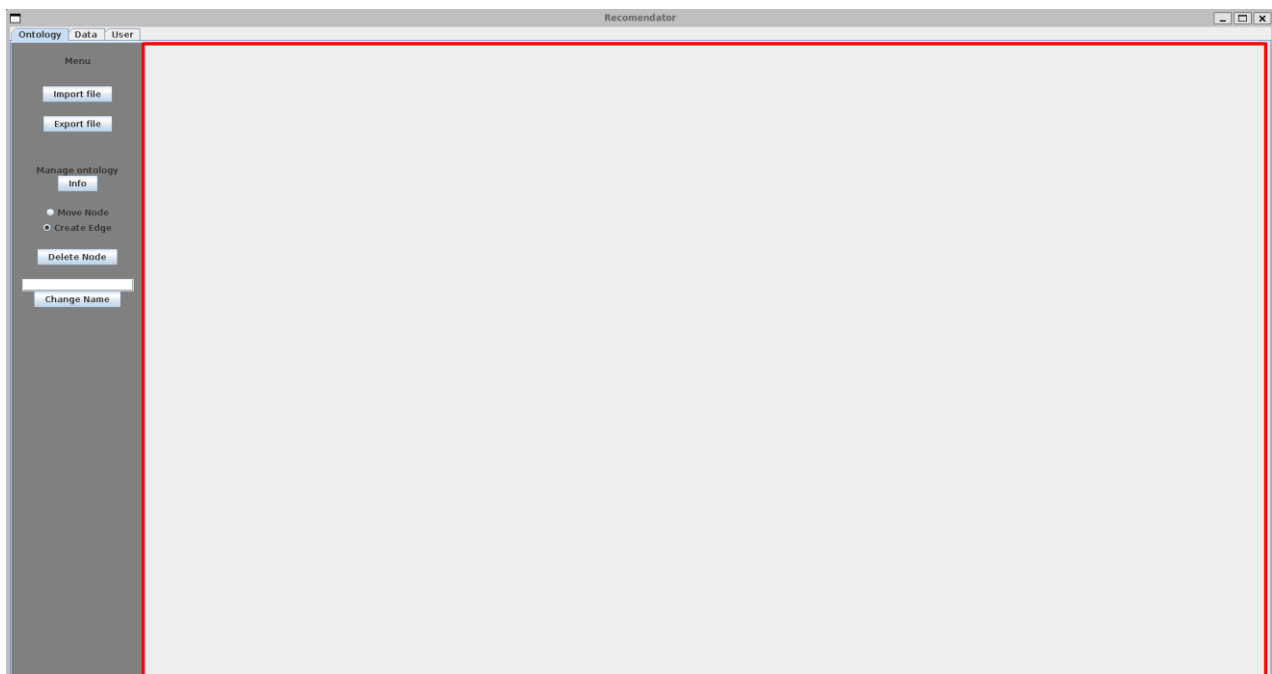
Un cop ja tenim els elements remarcats, anem a enumerar-los i explicar les diferents funcionalitats que tenen:

1. Administrador de finestres: petit administrador de finestres situat a la part superior esquerra de la interfície, aquest ens permet canviar entre les diferents finestres que componen el nostre programa, actualment només tenim la finestra d'ontologies i la finestra de dades.
2. Menu interactiu: menu situat a la part esquerra de la interfície que permet realitzar diferents funcions mitjançant botons, enumeram cada un d'aquests:
 - 2.1. Importar arxíu -> el primer element és un botó que ens permet importar arxius d'ontologies ja creades per carregar-les dins el programa i que es visualitzin automàticament.
 - 2.2. Exportar arxíu -> el segon element serveix per quan ja tens una ontologia pròpia feta, que la puguis exportar en forma d'arxíu per així tenir-la guardada i que en un futur la puguin modificar i tornar-la a carregar.
 - 2.3. Mostrar instruccions -> botó que quan es pitja, visualitza una finestra explicant les principals funcions que pot fer aquesta pantalla.
 - 2.4. Radio Buttons -> el primer element està compost per 2 [radio buttons](#)[25], així que només podem tenir una opció escollida a la vegada, per defecte tenim seleccionat l'opció de crear arestes entre els nodes, l'altre opció és la de desplaçar el node.

- 2.5. Borrar node -> últim element del menú que ens permet borrar el node que tinguem seleccionat al moment, barrant així totes les seves relacions amb altres nodes.
 - 2.6. Modificació nom -> últim element del menú que ens permet canviar el nom del node que tinguem seleccionat, una vegada tinguem el node seleccionat i el nou nom, basta pitjar el botó i es canviarà el nom del node
3. Marc de representació de les ontologies -> element central de la interfície, aquest marc representarà les ontologies que nosaltres importem o creem d'una forma senzilla i fàcil de comprendre.

6.3 Visualització

En aquest apartat, explicarem la visualització que hem adaptat en el nostre projecte i la forma en que el programa és capaç de visualitzar i interpretar les ontologies.

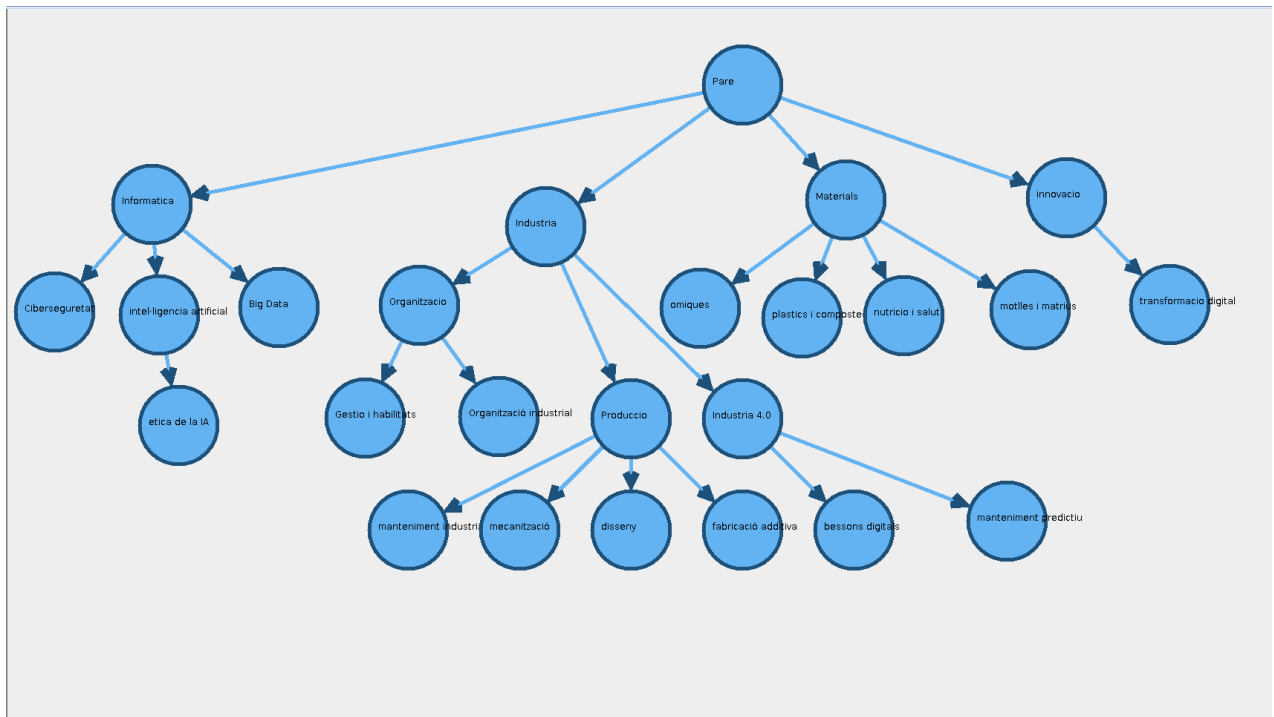


Imatge 11: Marc de representació de les ontologies

En la imatge superior, podem veure com és la interfície del nostre programa en l'apartat de les ontologies, és una estructura simple i orientativa. En el nostre cas, la representació constarà de un node "arrel", del qual penjaran tots els subnodes de l'ontologia, fent una [estructura d'arbre](#)[26], on cada node podrà tenir múltiples fills.

Entre els mateixos nodes hi pot haver relacions, que seran unidireccionals i no es podran crear cicles de forma que comprometi l'estructura de l'ontologia.

De la imatge cal destacar el requadre vermell, que és l'espai de representació de les ontologies dins del nostre projecte, és una mena de “canvas”, on es podran crear nodes, crear relacions entre els nodes, desplaçar els nodes i eliminar els nodes com vulgui l'usuari.



Imatge 12: **Exemple d'ontologia simple** dins del nostre programa.

Aquí podem veure com queda el nostre disseny de representació d'ontologies. Un cop hem visualitzat l'ontologia de forma gràfica, es pot comprendre molt millor la informació i la seva interpretació.

Com hem dit abans, la representació està basada en un arbre, tenim un node arrel, i d'ell penja la resta de nodes que conformen l'ontologia. En el nostre cas tenim 3 tipus possibles de nodes:

- Node “arrel” -> el node que està a la part superior de la nostra jerarquia de nodes, normalment no aporta cap informació útil, simplement és un node de suport que manta unida l'ontologia.
- Nodes intens -> s'entén com el node que té un o diversos fills que contenen informació més específica que la informació que aporta aquest i no són l'arrel de l'ontologia.

- Nodes fulla -> el node que té un o diversos pares, la seva funció és especificar la informació que contenen els seus respectius pares, normalment el node fill també pot ser un node “fulla” si aquest no té cap fill.

Cal destacar que un node pot anar canviant segons com es modifiqui l'ontologia

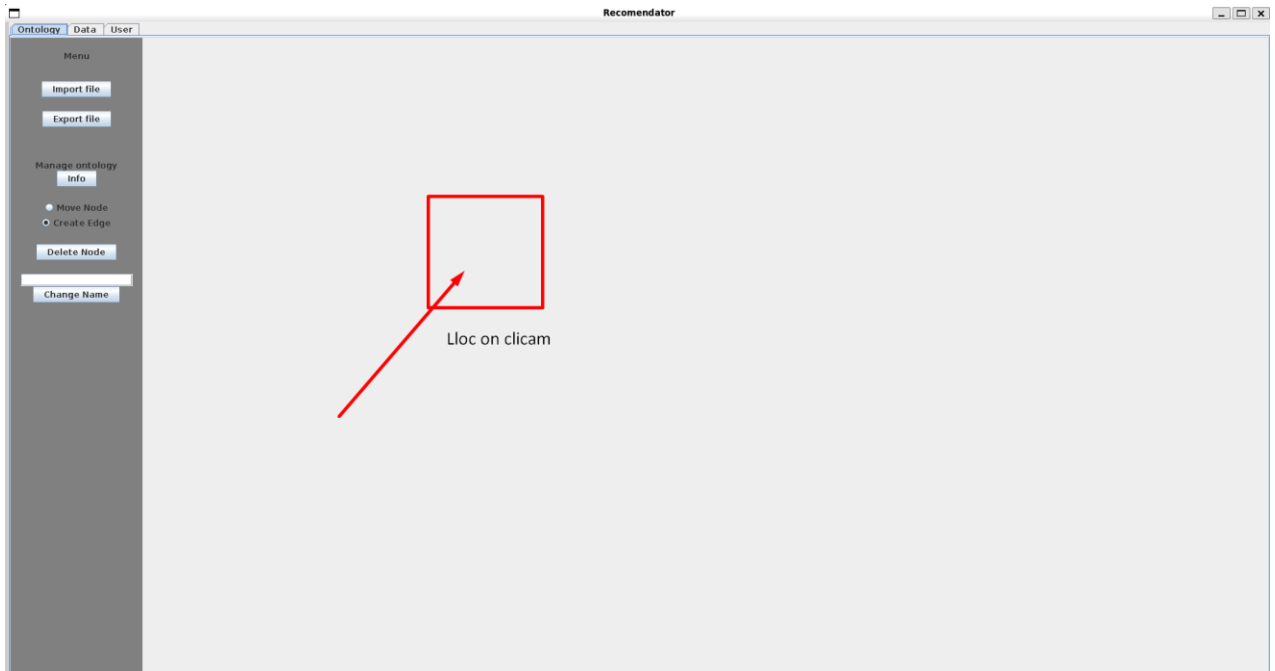
6.4 Creació ontologies

Una vegada ja sabem la forma de representació de les ontologies, ara explicarem les diverses formes que tenim de crear una ontologia en el nostre programa.

6.4.1 Nova ontologia

Aquest cas seria crear una ontologia des de 0, en el nostre programa hem aplicat una funció que si cliques dues vegades a la zona on es visualitzen les ontologies, apareixerà una finestra que et demana que introduequis el nom del node que vols crear. Aquesta no desapareixera fins que creis el node o cancelis l'operació de creació, cal dir que no podràs realitzar cap altra acció mentres hi hagi aquesta finestra oberta.

Aquesta acció es pot realitzar les vegades que es vulguin fins a crear una ontologia completa. El primer pas serà escollir el punt on volem crear el node, un cop escollit, clicarem dues vegades per iniciar el procés de creació d'un nou node.



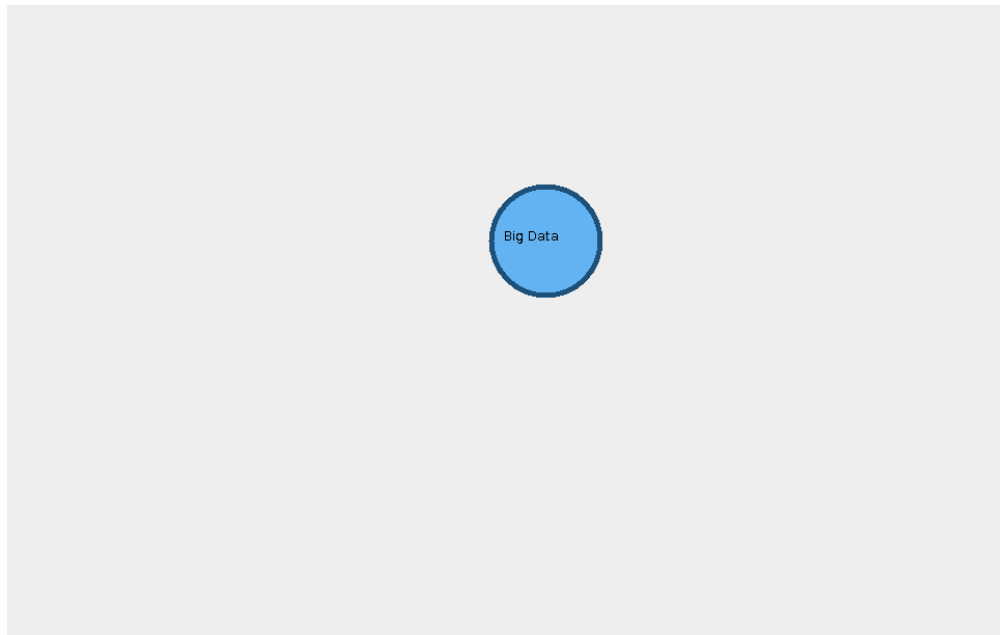
Imatge 13: **1r pas en la creació d'un nou node**: escollir les seves coordenades. Una vegada completat el 1r pas, seguirem amb el següent, que bàsicament consistirà escollir el nom del node que volem.



Imatge 14: **2n pas en la creació d'un node**: escollir el nom.

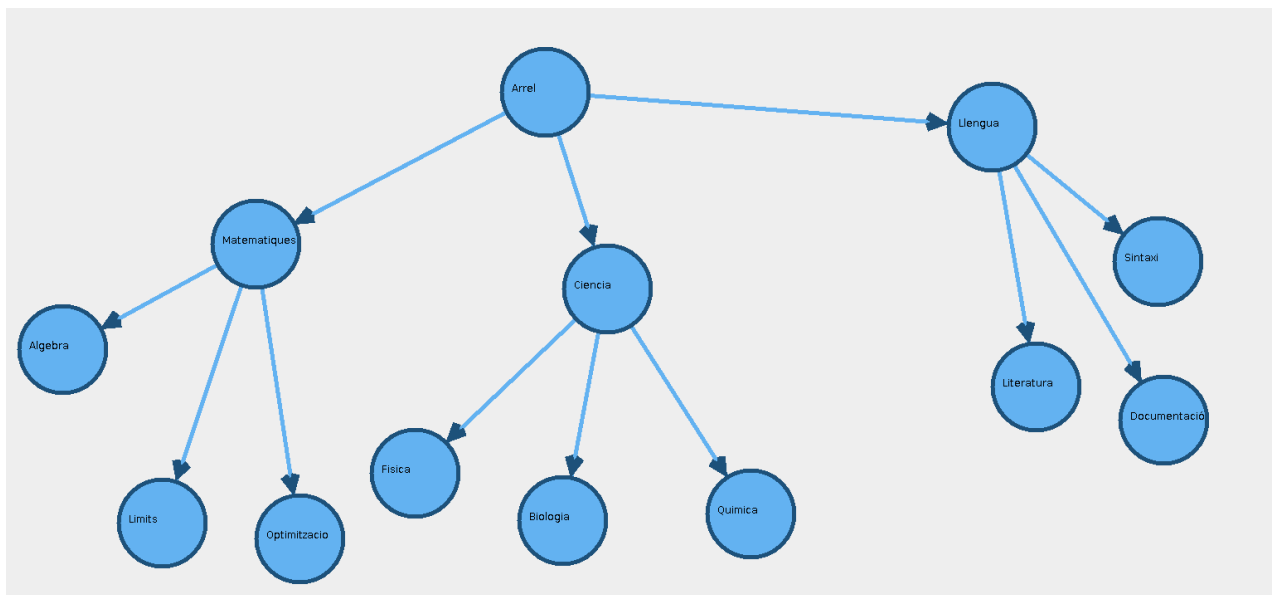
Un cop haguem fet els 2 passos, ja hurem acabat amb el procediment de crear un nou node dins de la nostra ontologia, cal dir que el programa no admet nodes sense nom.

Una vegada aquí, només hauríem de crear nous nodes i crear relacions entre ells per construir una ontologia senzilla.



Imatge 15: **Resultat del procediment de creació** d'un nou node.

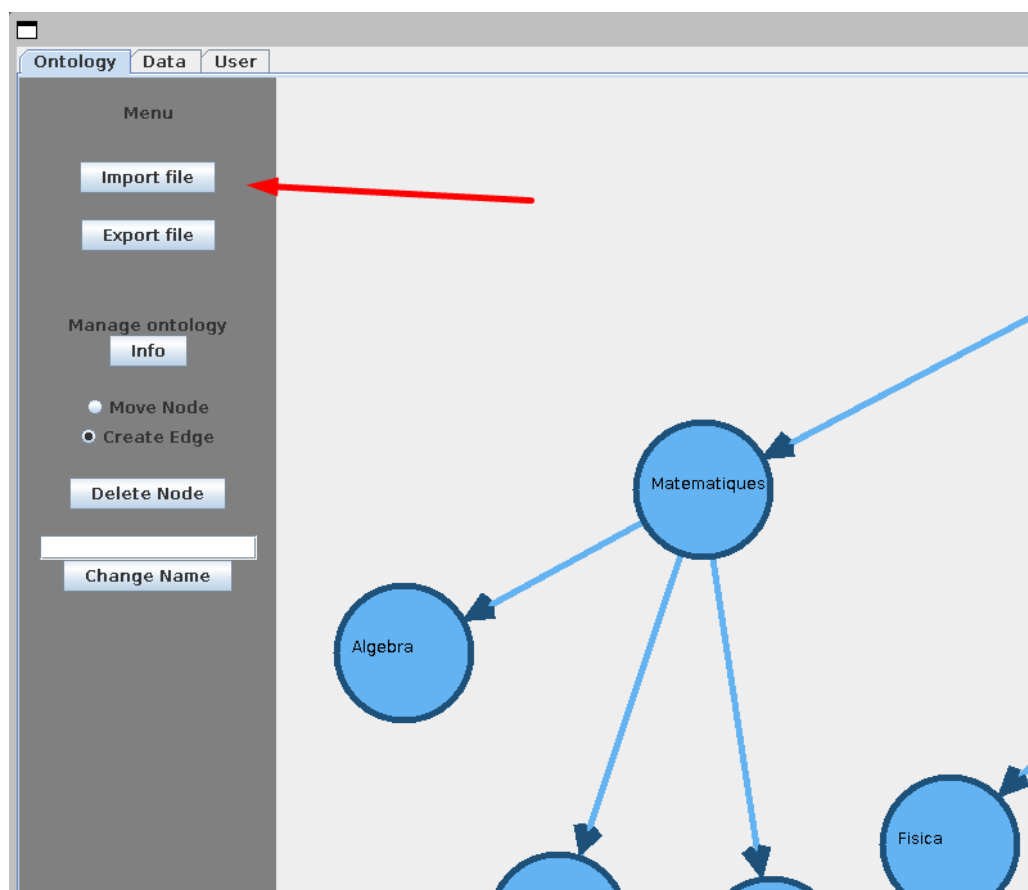
Aquesta forma de crear les ontologies és més lenta, però ofereix un grau de llibertat que l'altre forma no ofereix, aquí, a mesura que vas creant nodes i relacions, pots anar triant el patró de dibuix de l'ontologia, de l'altre forma ja et donen l'ontologia dibuixada, i moure els nodes un per un pot arribar a ser molest.



Imatge 16: **Exemple d'ontologia completa creada a partir de 0.**

6.4.2 Importar ontologia

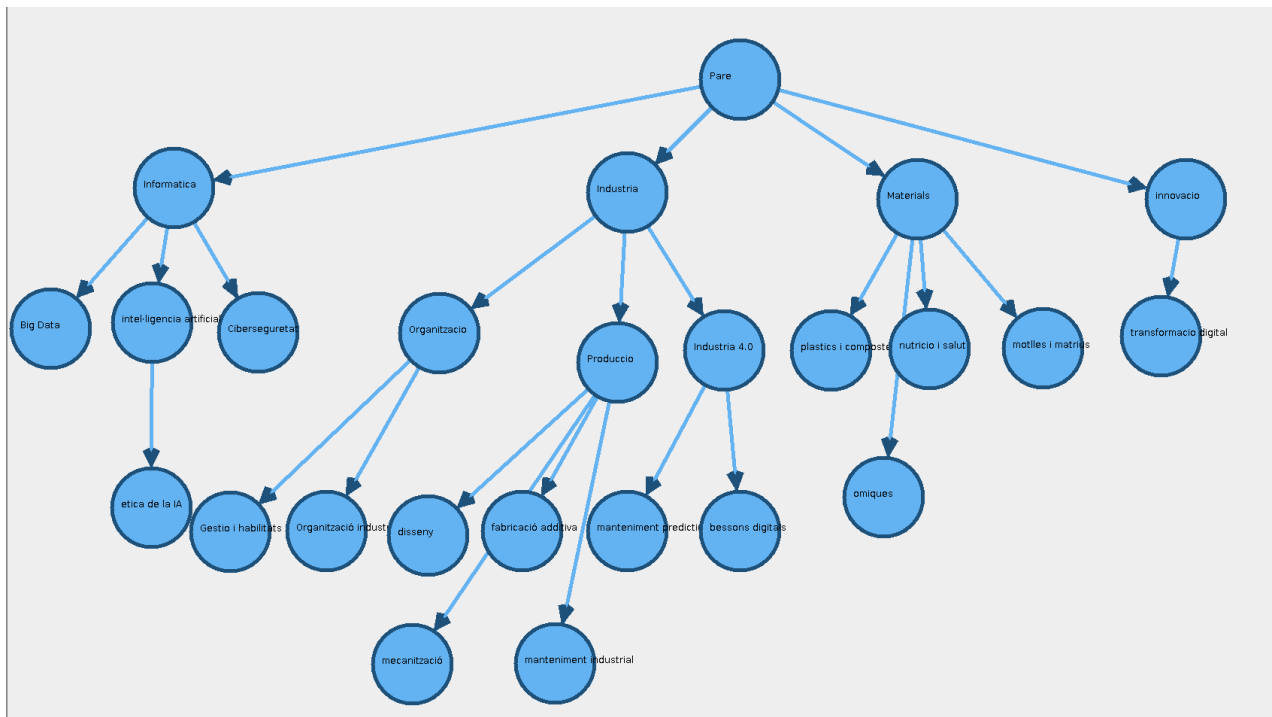
L'altre cas per crear una ontologia, és importar-ne una de feta. El programa té una funció que permet importar ontologies ja creades mitjançant un fitxer de text qualsevol. Per realitzar aquest procediment, s'ha de clicar al botó d'importar ontologies. Cal mencionar que si tenim una ontologia fent-se i no s'ha guardat, aquesta es perdria per complet.



Imatge 17: **Botó on s'ha de clicar per importar** una ontologia a partir de un fitxer de text.

Un cop hem seleccionat el fitxer que volem, al programa generarà l'estructura d'arbre de nodes automàticament. Creant els nodes i les relacions com s'indica al fitxer, el programa començarà amb el node arrel, indispensable que estigui en el fitxer i anirà dibuixant els nodes mitjançant un [DFS](#)[27].

A la imatge següent podem veure el resultat d'importar el fitxer de la imatge 5, que hem mostrat abans, es pot veure clarament l'estructura jeràrquica que segueix la representació del node, començant des de el node arrel i avant baixant fins a arribar als nodes fulla.



Imatge 18: **Resultat d'importar l'ontologia** de la imatge 7.

Com hem mencionat a l'apartat anterior, cada forma de creació d'ontologies té els seus beneficis i les seves desavantatges.

En aquest cas, crear l'ontologia amb el fitxer de text és més ràpid que el mètode anterior, en canvi aquí la gran desavantatge és si la forma en que es representa el graf no ens acaba d'agradar, en aquest cas haurem de moure els nodes un per un, lo que en grafs amb un gran nombre de nodes pot ser una operació costosa a nivell temporal.

6.5 Edició d'ontologies

Finalment, en l'apartat de les ontologies, tenim les funcions per editar una ontologia.

Hem creat una sèrie de funcions que permeten modificar una ontologia com vengui de gust, aquí mencionarem el funcionament de cada una d'aquestes funcions.

- Crear un nou node -> aquesta ja l'hem mencionada abans, si es clica dos cops en un espai on no hi ha cap node, ens sortira una finestra que ens demanarà el nom del nou node que volem crear.

- Eliminar node -> a l'igual que podem crear, també tenim l'opció d'eliminar un node de la nostra ontologia si no ens agrada, un cop s'elimina el node, totes les relacions que tenia amb altres nodes també s'eliminaran, per poder eliminar un node:
 1. Seleccionar el node que volem eliminar -> si clicam damunt un node, aquest canviarà de color, com podem veure a la imatge 6, això vol dir que aquest és el node que tenim seleccionat en aquest moment.
 2. Clicam el botó d'eliminar node i l'ontologia s'actualitzarà automàticament.

- Crear un arc entre dos nodes -> amb aquesta, podem crear una relació pare-fill entre dos nodes. Per a poder realitzar aquesta funcionalitat, haurem de tenir seleccionat el "radio button" de Crear Aresta. Les passes per realitzar això són:
 1. Seleccionar el node pare-> clicam damunt el node desitjat
 2. Mantenir clicat sobre el node pare, i arrastrar la fletxa fins al node que volem que sigui el seu "fill". Mentre mantinguem clicat, sortirà una línia des de el node pare, que ens facilitarà seleccionar el node que volem com a "fill". Si la línia està damunt el node desitjat, serà tan fàcil com deixar de clicar.

- Desplaçar un node -> ens servirà per si volem canviar d'ubicació un node, millorant així la visibilitat de l'ontologia si no ens agrada o si hi ha nodes que es trepitgen parcialment. En aquest cas, igual que l'anterior, el botó de Desplaçar un Node, haurà d'estar seleccionat. Les seves passes són:
 1. Seleccionar el node que volem desplaçar, igual que en el pas anterior.
 2. Mantenir clicat i desplaçar a la posició desitjada, un cop arribam a la posició, deixar de clicar, i el node es mourà. Mentre desplacen el node, es visualitzarà un nou node que ens servirà de referència per saber com quedarà el node desplaçat. Cal mencionar que les relacions amb altres nodes s'actualitzaran automàticament.

Com es pot veure, hem creat les funcions bàsiques de modificació. Qualsevol usuari podrà modificar l'ontologia sense dificultat.

De l'edició cal destacar un impediment, i és la limitació que els nodes només poden tenir un pare, es va aplicar aquesta limitació per facilitar la comprensió de l'ontologia i per distingir bé les diferents branques que pot tenir aquesta.

A més degut a aquesta limitació, evitam la creació de cicles dins de l'ontologia, cosa que podia causar problemes.

6.6 Funcionament intern

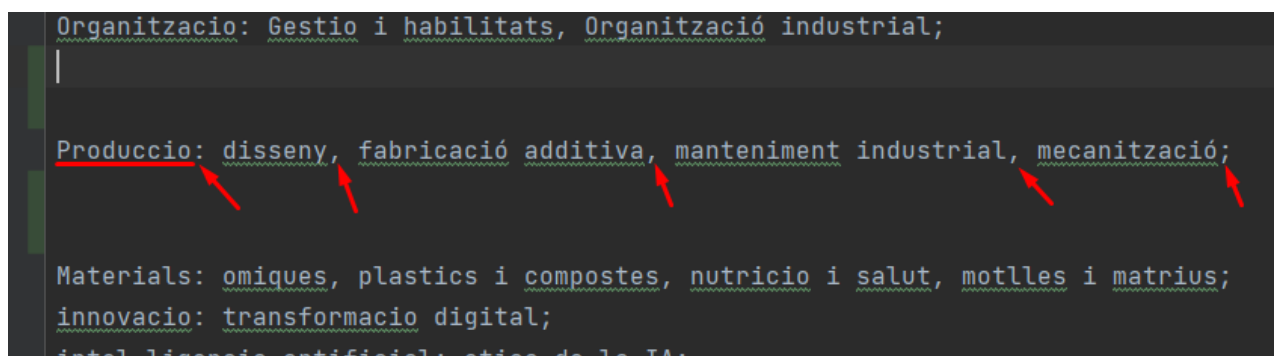
Una vegada s'ha explicat tot relacionat amb la pantalla d'ontologies, profunditzarem un poc en el seu funcionament, la forma en que cream nodes, el funcionament del marc de representació de les ontologies, la forma de dibuixar les ontologies que importam...

6.6.1 Creació de nodes

Igual que tenim 2 formes de crear ontologies, també tenim 2 formes de crear nodes, ja sigui manualment clicant dues vegades a un espai lliure al marc de representació o important una ontologia, on els nodes es crearan segons l'ontologia importada.

En el primer cas quan s'ha introduït el nom del nou node que volem crear, se li assigna un id automàticament i es crea el nou node, en aquest cas sense pare i sense relacions amb altres nodes. Quan es creï una relació amb un altre node, si aquest altre node actua com a pare, es crearà la relació i se li assignarà com a pare el node que ha establert la relació, a més els seus ancestres passaran a ser els ancestres del node pare més el nom del node pare. En l'altre cas, que el node creat faci de pare, els nodes que facin de fills s'afegiran al conjunt de nodes fills.

En el segon cas és més complexe, el que feim es llegir caràcter per caràcter i a mesura que ens anem trobant diferents signes de puntuació, sabem el que hem de fer, com hem mencionat abans quan ens trobem els dos punts (:), ens indica que hem acabat de llegir un node pare, i per tant ara passarem a llegir els nodes fills del node que acabam de llegir, separats cadascun per una coma (,) i que els fills del node acaben quan ens trobem el punt i coma (;) indicant-nos que el següent que llegirem serà un altra node pare.



```
Organitzacio: Gestio i habilitats, Organització industrial;  
Produccio: disseny, fabricació additiva, manteniment industrial, mecanització;  
Materials: omiques, plastics i compostes, nutricio i salut, motlles i matrius;  
innovacio: transformacio digital;  
intel·ligencia artificial: etica de la IA;
```

Imatge 19: Exemple de lectura d'ontologia.

Com hem dit abans, pels nodes fulla no importa crear una nova línia al fitxer, i això és degut al fet que a mesura que anem llegint els noms dels nodes, si aquests no estan creats, ja els cream,

és a dir, en l'exemple de la imatge d'adalt, un cop hem acabat de llegir la línia, tenim tots els noms dels nodes guardats en un array, iterem aquest array i si el node no està creat, el cream directament, per això un node fill es crea automàticament, sense haver de crear una nova línia per ell, que òbviament si es vol es podria crear una nova línia amb el seu nom i seguit de punt i coma (;), però el resultat no variaria.

6.6.2 Visualització de l'ontologia

Una vegada ja sabem el com es creen els nodes dins de l'ontologia, anem a veure el que ha estat el més complicat de realitzar del treball, i és la visualització de les ontologies quan s'importa un fitxer d'ontologies. On s'ha hagut d'utilitzar diversos mètodes de programació com DFS i [recursivitat](#)[28].

Una vegada s'ha llegit el fitxer es manda una ordre per pintar l'ontologia mitjançant una classe de Java base anomenada [canvas](#)[29], nosaltres hem creat la nostra pròpia classe que expandeix la classe mencionada anteriorment i l'hem anomenada Kanvas, aquesta apart de les funcions pròpies de canvas, també té funcions per comunicar-se amb la classe Graph per poder representar correctament els nodes. Una vegada es manda a pintar a l'ontologia, el que s'ha de fer és primer és trobar el node arrel, per això s'ha implementat una funció que itera a través dels nodes i el que no tengui pare, pues és el node arrel.

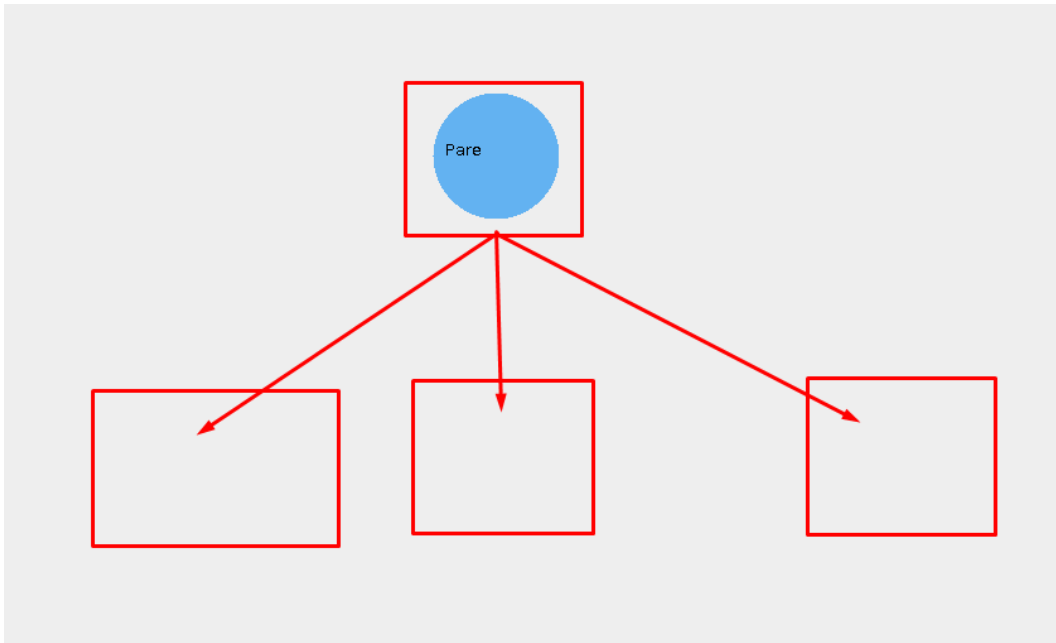
Una vegada ja tenim l'arrel de l'arbre de nodes, pintam el node en una posició preprogramada, en el nostre cas hem triat fer-ho a la posició (100,100). I a partir d'aquí el que feim és aplicar un DFS per anar pintant els nodes.

Tenim una [pila](#)[30] inicialment buida i quan obtenim el node arrel de l'ontologia, ficam el node dins la pila i iterarem la pila fins que aquesta estigui buida.

1. Agafam el node que estigui al principi de la pila i el treim.
2. Agafam tots els fills i els ficam a la pila.
3. Pintam els nodes fill en funció de la posició del pare

Com tenim una estructura en forma d'arbre, sempre voldrem dibuixar cap abaix del node pare per això tenim un HashMap de posicions, on guardam com a clau el id del node i com a valor la posició on aquest està pintat.

Com hem dit, pintarem els fills en funció del pare sempre cap avall, una vegada trobem una posició vàlida buida, afegim el node pintat al hashMap de posicions, però el complicat està quan volem pintar un node en una posició que ja està ocupada.



Imatge 20: **Visualització de com pintam els nodes.**

El principal problema que vam tenir a l'hora de fer aquesta secció va ser quan dos nodes es solapen un damunt l'altre i per tant en la visualització només es veu un dels nodes.

Per solventar aquest problema vam crear una funció que busca posicions alternatives vàlides per pintar un node.

El que feim bàsicament és primer mirar als costats de la posició on no s'ha pogut pintar el node. Si ja ens anem massa enfora de la posició de referència del node pare, baixam un nivell vertical.

D'aquesta forma aconseguim que l'arbre quedi estructurat i uniforme, sense tenir nodes molts dispersos.

7. Usuari

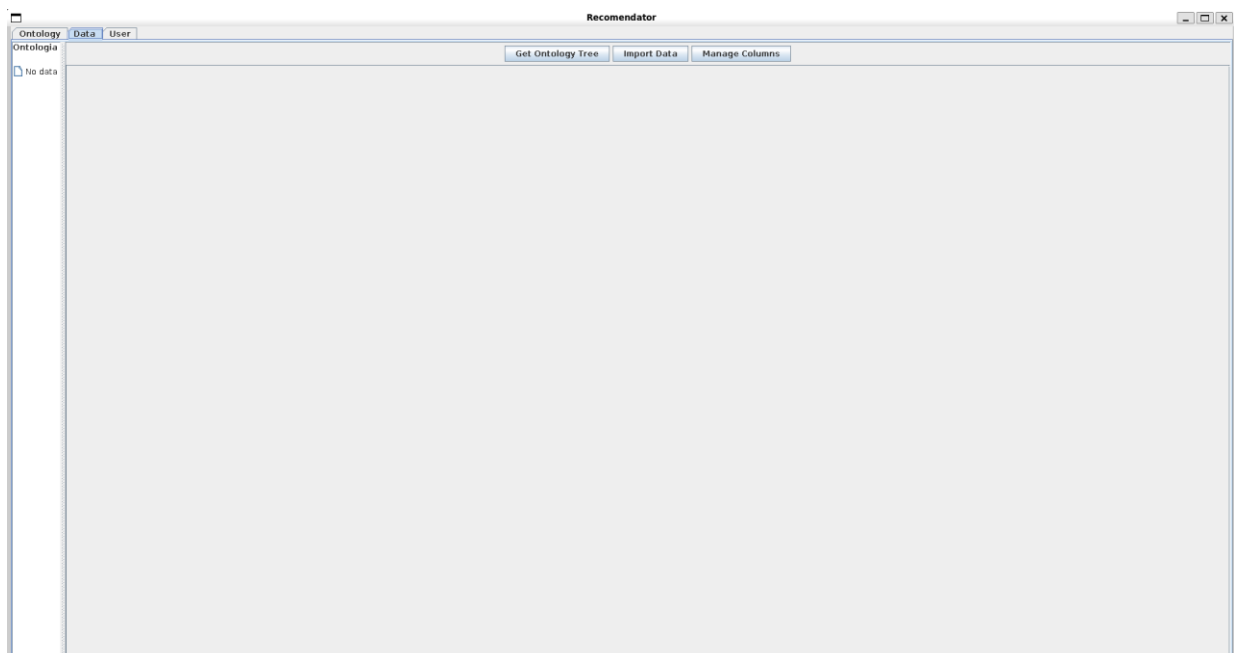
Fins ara hem vist tota la part d'ontologia i el repositori de cursos que involucra el nostre programa, ara que ja més o menys l'hem acabada d'explorar, anem a veure la part de l'usuari i de les dades donades per realitzar la recomanació personalitzada.

Aqui, igual que en els apartats anteriors, explicarem el procés de creació de tota la part de la recomanació, ja sigui la forma d'emmagatzemar i tractar la base de dades donada, les relacions que podem establir amb la part d'ontologia, la forma que tindrem d'obtenir les dades de l'usuari i un cop obtingudes, procedir a realitzar la recomanació.

7.1 Pantalla de dades

Ara ensenyarem els diferents components que formen part de la pantalla de dades del nostre programa, explicarem la seva funcionalitat i la seva forma d'actuar.

Primer de tot, per arribar a la pantalla haurem de clicar la finestra de Data a l'administrador de finestres situat a la part superior esquerra.

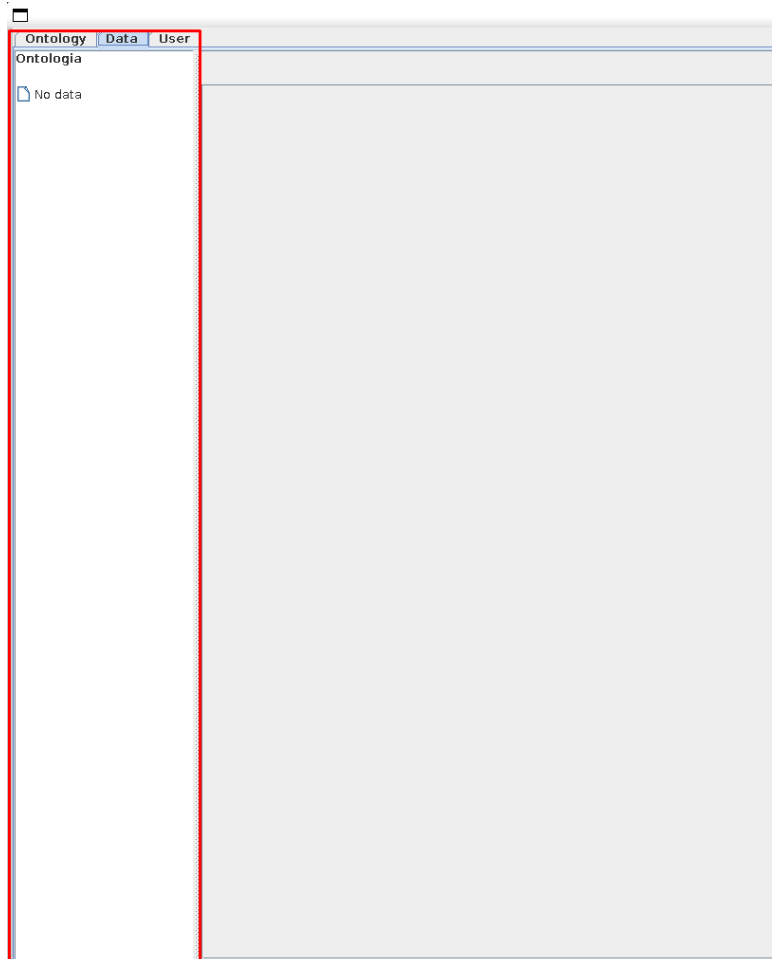


Imatge 21: **Pantalla de tractament de les dades.**

En aquesta imatge, visualitzem la que és la nostra pantalla per tractar els cursos i realitzar les recomanacions per a un usuari. S'ha dividit la pantalla principal en sub pantalles per diferenciar

bé cada funcionalitat, d'aquesta forma queda una pantalla molt més estructurada, organitzada i entensible per els usuaris. Igual que la pantalla de visualització d'ontologies, ens hem decantat per un estil minimalista, amb poca càrrega visual i colors freds, per no saturar la vista de l'usuari i no agotar-lo visualment.

Ara realitzarem una descripció detallada de cada subdivisió de la pantalla i dels elements que hi formen part.

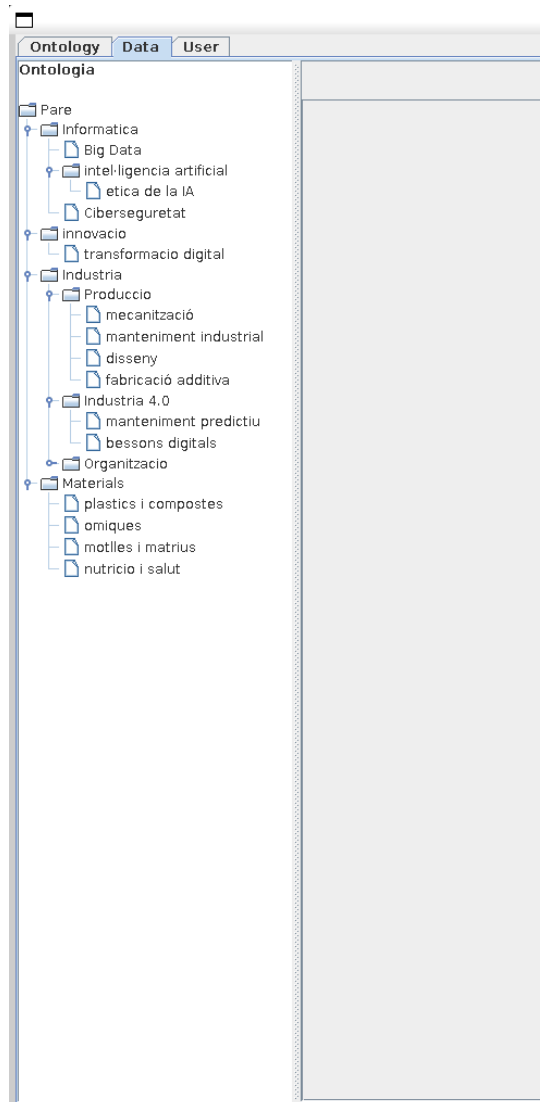


Imatge 22: **Panell lateral d'ontologies.**

El primer element que destacarem sobre la pantalla de dades és un panell lateral que té connexió amb l'ontologia que tinguem al moment, així podem visualitzar l'ontologia que tenim sense haver de canviar de pantalla, facilitant la visualització de les dades.

Per activar la funcionalitat és necessari que hi hagi una ontologia creada, si no hi ha cap ontologia ni cap node, només es visualitzarà com està a la imatge.

Si tenim una ontologia creada i activam la funcionalitat, es mostrarà de la següent manera:

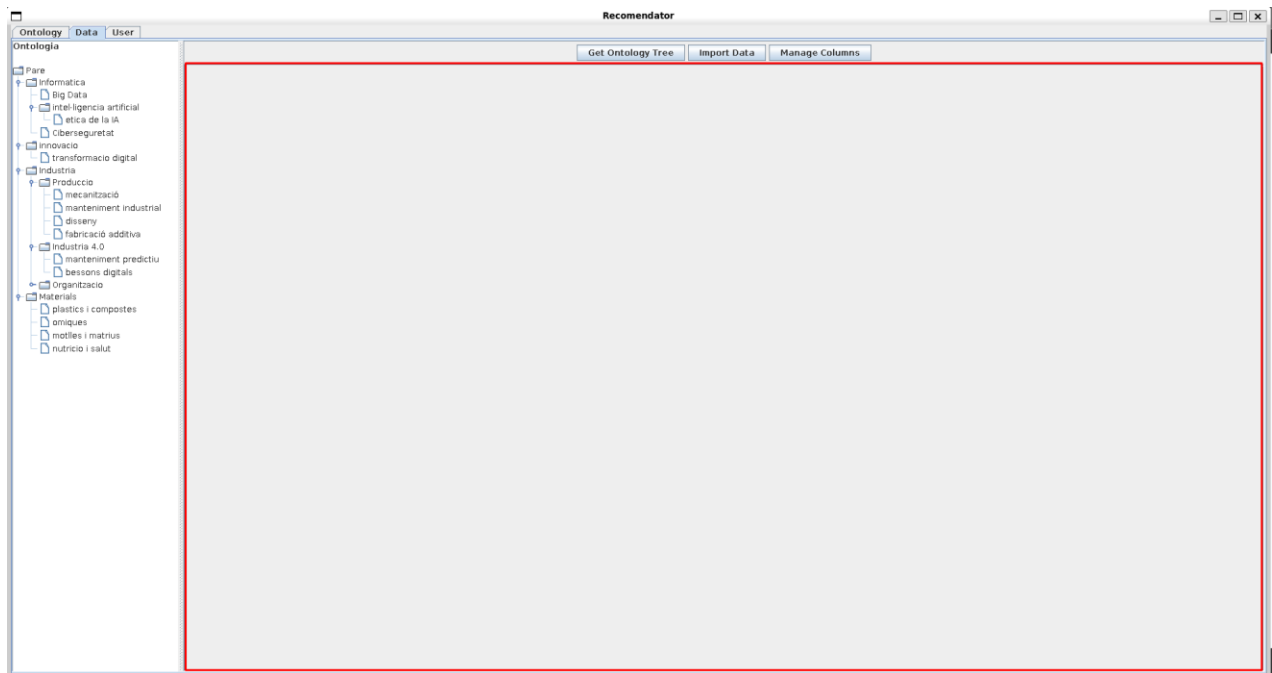


Imatge 23: **Panell lateral amb ontologia creada.**

Com es pot veure, recream en forma d'arbre l'ontologia que tenim carregada a la pantalla d'ontologies. A més aquest panell és dinàmic i interactiu, es pot expandir i contreure els nodes fills de qualsevol node pare, per si es vol mostrar tot l'arbre o només una part d'aquest.

A més si l'ontologia és modifica, si tornam a importar l'arbre de l'ontologia, aquest s'actualitzarà segons com estigui l'ontologia al moment.

El següent element a destacar de la pantalla de dades és el panell de la base de dades de cursos dels quals farem recomanació:



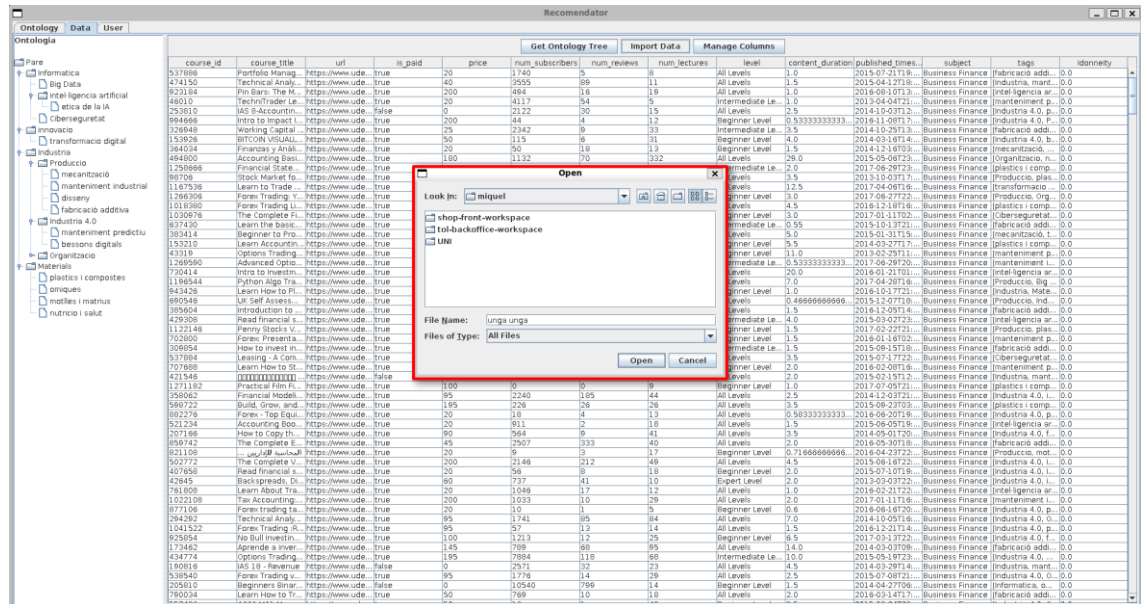
Imatge 24: **Panell de visualització de la base de dades.**

Al iniciar l'aplicació, aquest panell estarà en blanc, ja que no tindrem cap base de dades al sistema i el nostre programa no sap ni quines columnes té la base de dades ni les files que hi haurà.

Per tant, al no tenir una estructura predefinida, podem carregar qualsevol tipus de base de dades, el més important i indispensable de moment és que aquesta base de dades ha tenir una columna anomenada "Tags" per poder realitzar les recomanacions.

Per això, quan importem la base de dades al programa, aquesta pantalla s'actualitzarà al moment i ens mostrarà les dades que haguem importat:

- 1.2 Passam ara al botó d'importació de dades, la seva funció la diu el mateix nom, quan cliquem aquest botó ens sortirà una finestra que ens demanarà quin arxiu de dades volem obrir per carregar les dades que conté.



Imatge 28: Pantalla per escollir l'arxiu de dades.

Els arxius han de ser un tipus que contengui les dades classificades en forma d'excel, com per exemple arxius amb l'extensió .csv .

Una vegada s'hagin carregat les dades, el panell de visualització passarà de la imatge 25 a la imatge 26, ensenyant les files i columnes de la base de dades, rellenant el panell per complet.

- 1.3 Arribam al darrer botó de la pantalla de dades, amb aquest es pot elegir quines columnes visualitzar al moment. Al clicar el botó, sortirà una finestra nova amb una llista de [checkbox](#)[31] cadascuna tindrà el nom d'una de les columnes de la base de dades. Si la casella està marcada, indica que la columna es visualitzarà al panell de dades. Una vegada hem escollit quines columnes volem visualitzar, bastarà clicar al botó de "Ok" per actualitzar la vista del panell i només visualitzar les columnes que quedin marcades.

logy	Data	User	Get Ontology Tree										Import Data	Manage Columns
	course_id	course_title	url	is_paid	price	num_subscribers	num_reviews	num_lectures	level	content_duration	published_times...	subject	tags	idonnety
informatica	537898	Portfolio Manag...	https://www.ud...	true	20	1740	5	8	All Levels	1.0	2015-07-2119...	Business Finance	Fabricació addi...	0.0
Big Data	474150	Technical Analy...	https://www.ud...	true	40	3555	89	11	All Levels	1.5	2015-04-2718...	Business Finance	Indústria, mant...	0.0
inteligencia artificial	623184	Pin Bars: The M...	https://www.ud...	true	200	484	16	19	All Levels	1.0	2016-08-1013...	Business Finance	Inteligència ar...	0.0
etica de la IA	46012	TechnTrader Le...	https://www.ud...	true	20	4117	24	5	Intermediate Le...	0.0	2015-04-0472...	Business Finance	Manteniment p...	0.0
obsequeretat i innovació	253810	IAS 8-Accountin...	https://www.ud...	false	0	2122	30	15	All Levels	2.5	2014-10-03712...	Business Finance	Indústria 4.0, p...	0.0
transformació digital	994686	Intro to Impact I...	https://www.ud...	true	200	44	4	12	Beginner Level	0.5333333333333333	2016-11-08717...	Business Finance	Indústria 4.0, p...	0.0
Indústria	529646	Working Capital	https://www.ud...	true	25	2242	9	33	Intermediate Le...	5.5	2014-10-25713...	Business Finance	Fabricació addi...	0.0
Producció	153926	BIFCOIN VISUALL...	https://www.ud...	true	50	115	6	31	Beginner Level	4.0	2014-03-16714...	Business Finance	Indústria 4.0, B...	0.0
mecanització	364034	Finanzas y Análisi...	https://www.ud...	true	20	90	18	13	Beginner Level	1.5	2014-12-16703...	Business Finance	Mecanització, i...	0.0
Manteniment Industrial	484800	Accounting Basis...	https://www.ud...	true	180	1132	70	332	All Levels	28.0	2015-05-08713...	Business Finance	Organizació, m...	0.0
dissecció	1258666	Financial State...	https://www.ud...	true	50	0	0	9	Intermediate Le...	2.0	2017-06-29723...	Business Finance	Plàstics i compo...	0.0
fabricació additiva	96706	Stock Market fo...	https://www.ud...	true	35	143	15	42	All Levels	3.5	2013-10-03717...	Business Finance	Producció, plàs...	0.0
Indústria 4.0	1187536	Learn to Trade	https://www.ud...	true	195	89	7	64	All Levels	12.5	2017-04-06716...	Business Finance	Transformació ...	0.0
Manteniment predictiu	1266306	Forex Trading Y...	https://www.ud...	true	140	1026	25	37	Beginner Level	3.0	2017-06-27722...	Business Finance	Producció, Org...	0.0
bessons digitals	1018380	Forex Trading Li...	https://www.ud...	true	195	677	45	40	All Levels	4.5	2016-12-18719...	Business Finance	Plàstics i compo...	0.0
Organització	1030976	The Complete Fi...	https://www.ud...	true	150	50	0	0	Beginner Level	3.0	2017-01-11702...	Business Finance	Ciberseguret...	0.0
Materials	837430	Learn the basic...	https://www.ud...	true	20	0	0	0	Intermediate Le...	5.55	2015-10-13721...	Business Finance	Fabricació addi...	0.0
plàstics i compostes	363414	Beginner to Pro...	https://www.ud...	true	150	0	0	0	All Levels	5.0	2015-01-31715...	Business Finance	Mecanització, t...	0.0
omiques	152120	Learn Accountin...	https://www.ud...	true	200	0	0	0	Beginner Level	5.5	2014-03-20717...	Business Finance	Plàstics i compo...	0.0
nutrició i salut	43319	Options Trading	https://www.ud...	true	100	0	0	0	Beginner Level	11.0	2015-02-26711...	Business Finance	Manteniment p...	0.0
	1269590	Advanced Optio...	https://www.ud...	true	40	0	0	0	Intermediate Le...	0.5333333333333333	2017-06-29720...	Business Finance	Manteniment I...	0.0
	730414	Intro to Investm...	https://www.ud...	true	60	0	0	0	All Levels	20.0	2016-01-21701...	Business Finance	Inteligència ar...	0.0
	1196544	Python Algo Tra...	https://www.ud...	true	200	0	0	0	All Levels	7.0	2017-04-26719...	Business Finance	Producció, Big ...	0.0
	643426	Learn How to Pl...	https://www.ud...	true	35	0	0	0	Beginner Level	1.0	2016-10-17721...	Business Finance	Indústria, Mate...	0.0
	660546	UK Self Assess...	https://www.ud...	true	20	0	0	0	All Levels	0.4666666666666666	2015-12-07713...	Business Finance	Producció, Ind...	0.0
	702800	Forex Presenta...	https://www.ud...	true	25	0	0	0	All Levels	1.5	2016-12-05714...	Business Finance	Fabricació addi...	0.0
	429300	Read financial s...	https://www.ud...	true	50	0	0	0	Intermediate Le...	4.0	2015-03-02718...	Business Finance	Inteligència ar...	0.0
	1122146	Penny Stocks V...	https://www.ud...	true	200	0	0	0	Beginner Level	1.5	2017-02-22721...	Business Finance	Producció, plàs...	0.0
	308954	How to invest in...	https://www.ud...	true	30	0	0	0	Intermediate Le...	1.5	2015-12-05714...	Business Finance	Fabricació addi...	0.0
	537884	Leasing - A Com...	https://www.ud...	true	20	3094	2	22	All Levels	3.5	2015-07-17722...	Business Finance	Ciberseguret...	0.0
	707668	Learn How to St...	https://www.ud...	true	50	2635	13	29	Beginner Level	2.0	2016-02-08716...	Business Finance	Manteniment p...	0.0
	421446	Practical Film F...	https://www.ud...	true	100	0	0	0	All Levels	2.0	2015-02-15712...	Business Finance	Indústria, mant...	0.0
	1271182	Practical Film F...	https://www.ud...	true	100	0	0	0	Beginner Level	1.0	2017-07-05721...	Business Finance	Plàstics i compo...	0.0
	350862	Financial Model...	https://www.ud...	true	90	2240	165	44	All Levels	2.5	2014-12-20721...	Business Finance	Indústria 4.0, ...	0.0
	569222	Build, Grow, and...	https://www.ud...	true	195	228	29	29	All Levels	3.5	2015-09-23703...	Business Finance	Plàstics i compo...	0.0
	862276	Forex - Top Equi...	https://www.ud...	true	20	18	4	13	All Levels	0.5833333333333333	2016-06-20719...	Business Finance	Indústria 4.0, p...	0.0
	521234	Accounting Boo...	https://www.ud...	true	20	911	2	18	All Levels	1.5	2015-06-05719...	Business Finance	Inteligència ar...	0.0
	207166	Forex Trading R...	https://www.ud...	true	100	594	9	41	All Levels	1.5	2014-09-01720...	Business Finance	Indústria 4.0, ...	0.0
	856742	The Complete E...	https://www.ud...	true	45	2507	333	40	All Levels	2.0	2016-05-30718...	Business Finance	Fabricació addi...	0.0
	821108	المحاسبة للأرباح وغيره...	https://www.ud...	true	20	9	3	17	Beginner Level	0.7166666666666666	2016-04-23722...	Business Finance	Producció, mot...	0.0
	502722	The Complete V...	https://www.ud...	true	20	2146	212	49	All Levels	4.0	2015-06-18722...	Business Finance	Indústria 4.0, ...	0.0
	407658	Read financial s...	https://www.ud...	true	20	56	8	18	Expert Level	2.0	2015-07-10719...	Business Finance	Indústria 4.0, ...	0.0
	42645	Backspreads, Di...	https://www.ud...	true	60	737	41	10	Expert Level	2.0	2013-03-03722...	Business Finance	Indústria 4.0, ...	0.0
	761808	Learn How to Pla...	https://www.ud...	true	20	1046	17	12	All Levels	1.0	2016-02-21722...	Business Finance	Inteligència ar...	0.0
	1022108	Tax Accounting...	https://www.ud...	true	200	1033	10	29	All Levels	2.0	2017-01-11716...	Business Finance	Manteniment p...	0.0
	877106	Forex trading ta...	https://www.ud...	true	20	10	1	5	Beginner Level	0.6	2016-06-16720...	Business Finance	Indústria 4.0, p...	0.0
	294292	Technical Analy...	https://www.ud...	true	95	1741	85	84	All Levels	7.0	2014-10-05716...	Business Finance	Indústria 4.0, ...	0.0
	1041522	Forex Trading R...	https://www.ud...	true	100	57	13	14	All Levels	1.5	2016-12-21714...	Business Finance	Indústria 4.0, ...	0.0
	925954	No Bull Investm...	https://www.ud...	true	100	123	12	25	Beginner Level	6.5	2017-03-13722...	Business Finance	Indústria 4.0, ...	0.0
	173462	Aprende a invertir	https://www.ud...	true	145	789	68	93	All Levels	14.0	2014-03-03709...	Business Finance	Fabricació addi...	0.0
	434774	Options Trading	https://www.ud...	true	100	7884	118	68	Intermediate Le...	10.0	2015-05-19723...	Business Finance	Indústria 4.0, ...	0.0
	180816	IAS 18 - Revenue	https://www.ud...	false	0	2571	32	23	All Levels	4.5	2014-03-29714...	Business Finance	Indústria, mant...	0.0
	1638247	Forex Trading U...	https://www.ud...	true	100	1796	174	76	All Levels	19.4	2017-07-26719...	Business Finance	Indústria 4.0, ...	0.0

Imatge 29: Finestra d'elecció de les columnes a visualitzar.

Com, podem veure en aquest cas, només hem deixat 8 columnes visibles de la taula, al pitjar el botó de “Ok”, varem com el panell de dades actualitzà les columnes visibles:

	course_title	is_paid	price	num_lectures	level	content_duration	subject	tags	idonnety
	Portfolio Management for ...	true	20	8	All Levels	1.0	Business Finance	Fabricació additiva, Pro...	0.0
	Technical Analysis tools fo...	true	40	11	All Levels	1.0	Business Finance	Indústria, mantenim...	0.0
	Pin Bars: The Most Powerf...	true	200	19	All Levels	1.0	Business Finance	Inteligència artificial, Ind...	0.0
	TechnTrader Leading Hyb...	true	20	5	Intermediate Level	1.0	Business Finance	Manteniment predictiu, M...	0.0
	IAS 8-Accounting Polices ...	false	0	15	All Levels	2.5	Business Finance	Indústria 4.0, plàstics i co...	0.0
	Intro to Impact Investin...	true	200	12	Beginner Level	0.5333333333333333	Business Finance	Indústria 4.0, fabricació a...	0.0
	Working Capital Managem...	true	25	33	Intermediate Level	3.5	Business Finance	Fabricació additiva, intel...	0.0
	BIFCOIN VISUALLY, Part I...	true	50	31	Beginner Level	4.0	Business Finance	Indústria 4.0, bessons di...	0.0
	Accounting Basics - A Com...	true	20	13	Beginner Level	4.0	Business Finance	Mecanització, motlles i matr...	0.0
	Financial Statement Audit...	true	180	332	All Levels	29.0	Business Finance	Organització, nutrició i sal...	0.0
	Stock Market for Beginne...	true	35	42	All Levels	3.5	Business Finance	Producció, plàstics i com...	0.0
	Learn How to Play CashFl...	true	20	16	All Levels	4.0	Business Finance	Transformació digital, int...	0.0
	Forex Trading: Your Compl...	true	140	37	Beginner Level	3.0	Business Finance	Producció, Organització, ...	0.0
	Forex Trading Like Banks ...	true	195	40	All Levels	4.5	Business Finance	plàstics i compostes, Mat...	0.0
	The Complete Financial St...	true	150	37	Beginner Level	3.0	Business Finance	Producció, Organització, ...	0.0
	Learn the basics of prep...	true	20	0	Intermediate Level	5.0	Business Finance	Ciberseguretad, Materia...	0.0
	Beginner to Pro in PowerF...	true	150	50	All Levels	5.0	Business Finance	Mecanització, transforma...	0.0
	Learn Accounting, Unders...	true	200	33	Beginner Level	5.5	Business Finance	plàstics i compostes, Mat...	0.0
	Options Trading Basics (3...	true	180	45	Beginner Level	11.0	Business Finance	Manteniment predictiu, O...	0.0
	Advanced Options Trading	true	40	6	Intermediate Level	0.5333333333333333	Business Finance	Manteniment industrial, f...	0.0
	Intro to Investment Bankin...	true	60	15	All Levels	20.0	Business Finance	Inteligència artificial, pla...	0.0
	Python Algo Trading: Sent...	true	200	42	All Levels	7.0	Business Finance	Producció, Big Data, Orga...	0.0
	Learn How to Play CashFl...	true	25	26	Beginner Level	1.0	Business Finance	Indústria, Materials, nutr...	0.0
	UK Self Assessment Tax R...	true	20	6	All Levels	0.46666666666666667	Business Finance	Producció, Indústria, man...	0.0
	Read financial statements...	true	25	8	All Levels	1.5	Business Finance	Fabricació additiva, Indu...	0.0
	Read financial statements...	true	50	38	Intermediate Level	4.0	Business Finance	Inteligència artificial, Inf...	0.0
	Forex Presentation and ...	true	20	22	Beginner Level	1.5	Business Finance	Manteniment predictiu, M...	0.0
	How to invest in Russian ...	true	30	12	Intermediate Level	1.5	Business Finance	Fabricació additiva, etica...	0.0
	Leasing - A Comprehensive...	true	20	22	All Levels	3.5	Business Finance	Ciberseguretad, manteni...	0.0
	Learn How to Start Small ...	true	50	29	Beginner Level	2.0	Business Finance	Manteniment predictiu, m...	0.0
	Practical Film Financing F...	true	0	25	All Levels	2.0	Business Finance	Indústria, manteniment pa...	0.0
	Financial Modeling: Build a...	true	95	44	All Levels	2.5	Business Finance	Indústria 4.0, intel·ligèn...	0.0
	Build, Grow, and Protect y...	true	195	26	All Levels	3.5	Business Finance	plàstics i compostes, Mat...	0.0
	Forex - Top Equity Manag...	true	20	13	All Levels	0.5833333333333333	Business Finance	Indústria 4.0, plàstics i co...	0.0
	Accounting Books of Acco...	true	20	18	All Levels	1.5	Business Finance	Inteligència artificial, Ind...	0.0
	How to Copy the Best Tra...	true	90	41	All Levels	3.5	Business Finance	Indústria 4.0, fabricació a...	0.0
	The Complete Ethereum C...	true	45	40	All Levels	2.0	Business Finance	Fabricació additiva, mant...	0.0
	المحاسبة للأرباح وغيره...	true	20	17	Beginner Level	0.71666666666666667	Business Finance	Producció, motlles i matr...	0.0
	Read financial statements...	true	20	16	All Levels	4.5	Business Finance	Indústria 4.0, Indústria, B...	0.0
	Backspreads, Diagonals a...	true	60	10	Expert Level	2.0	Business Finance	Indústria 4.0, intel·ligèn...	0.0
	Learn About Trading Optio...	true	20	12	All Levels	1.0	Business Finance	Inteligència artificial, Ind...	0.0
	Learn Accounting, Difficul...	true	200	29	All Levels	5.5	Business Finance	Manten	

Si tornéssim a clicar el botó, ja només ens surten marcades les columnes de la imatge anterior, d'aquesta forma, aconseguim una millor visualització de les dades, ocultant possibles columnes de les quals no ens interessa la seva informació en el moment.

7.2 Pantalla usuari

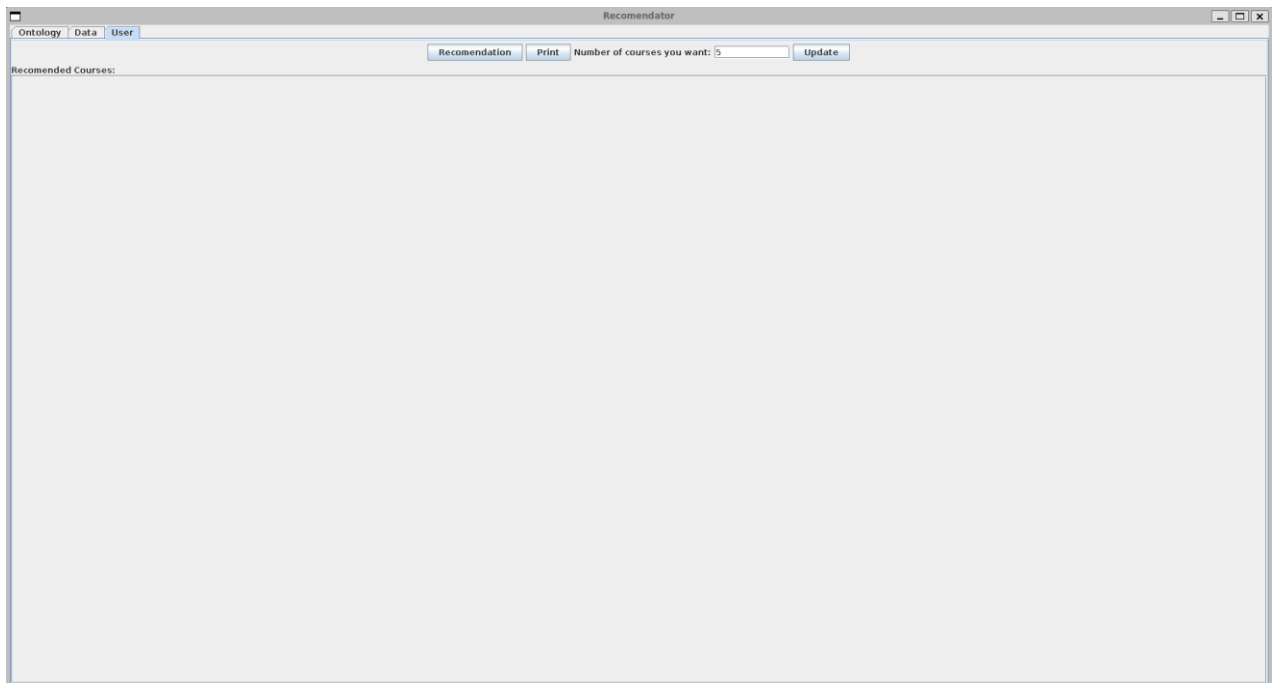
Passam a la pantalla final del nostre projecte, la pantalla de l'usuari, aquí, l'usuari haurà d'introduir els seus objectius que vol assolir amb la realització d'alguns dels cursos.

Aquesta pantalla, a diferència de les altres dues, necessita les dades carregades. Per a que aquesta pantalla sigui útil, les dades dels cursos i de l'ontologia ja han d'estar carregades al sistema. Si alguna no està ja carregada, aquesta pantalla no tindrà cap ús pràctic real, ja que no podrà funcionar adequadament degut a la falta d'informació.

Per això és important que abans d'accedir a aquesta, les altres dues ja hagin complert la seva funció de carregar en el sistema les dades necessàries per a realitzar una recomanació com toca.

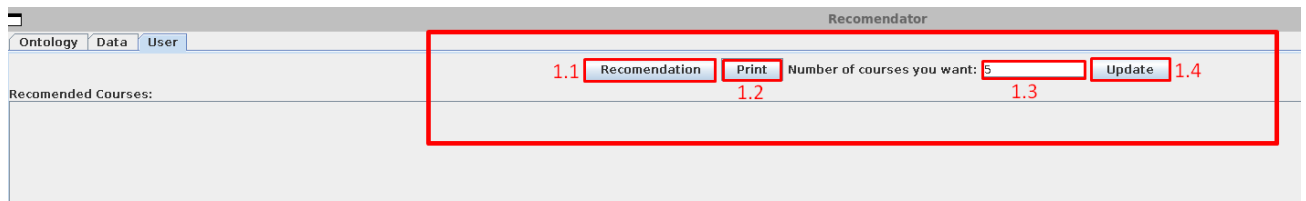
L'objectiu principal d'aquesta pantalla és obtenir les dades de l'usuari, els coneptes que vol aprendre, i realitzar una recomanació personalitzada dels cursos que més s'ajustin als seus criteris.

Finalment per arribar a la última pantalla, la de l'usuari, haurem de clicar sobre "User" en l'administrador de tasques situat a la part superior esquerra de la pantalla.



Imatge 31: **Pantalla de l'usuari.**

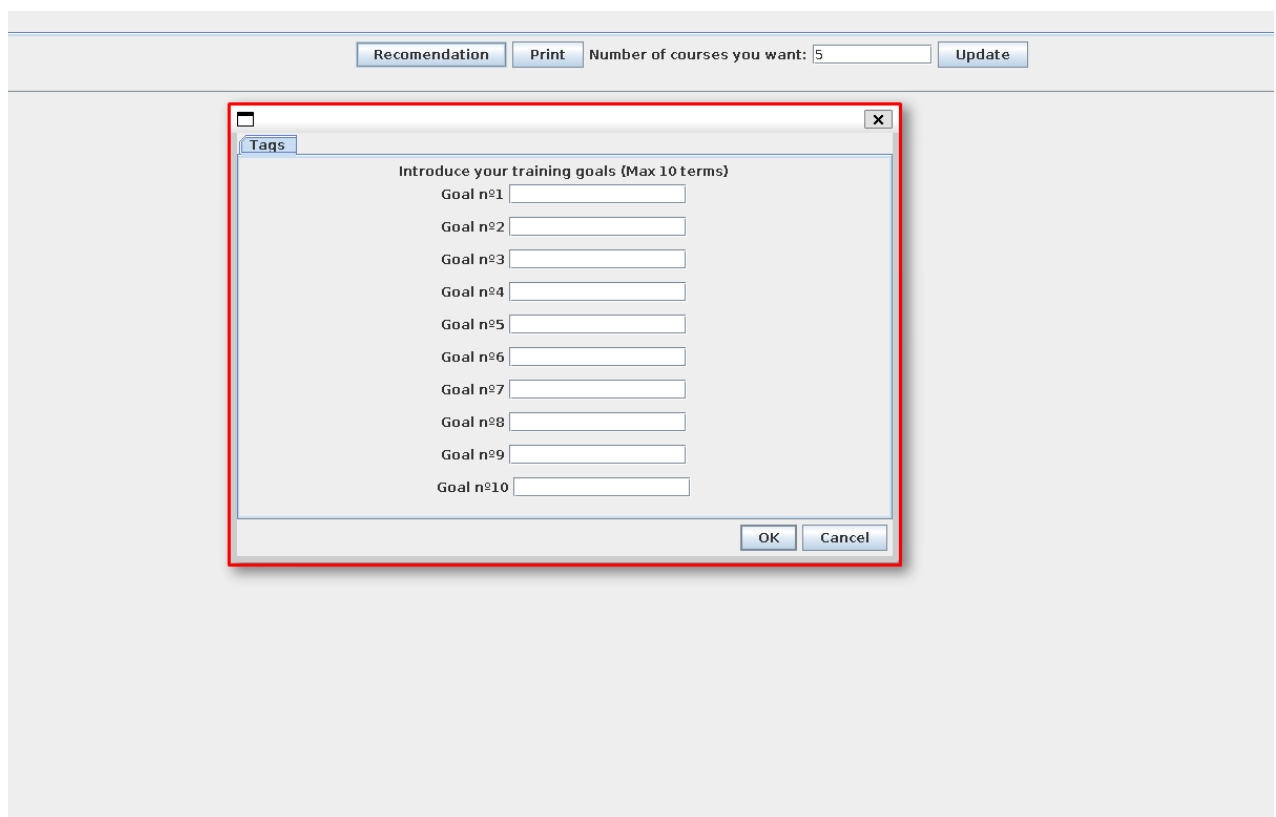
Com es pot veure, en aquesta pantalla igual que les altres té un estil simple i no molt carregat. El seu disseny és intuïtiu i fàcil d'usar, per tant els usuaris no hauràn de tenir experiència prèvia per a poder utilitzar aquesta pantalla ni ningú que els hi hagi d'ensenyar el seu funcionament. En aquesta pantalla hi ha 2 zones diferenciades, de les quals explicarem la seva funcionalitat i els seus diferents usos.



Imatge 32: Menú superior de la pantalla de l'usuari.

La primera part de la pantalla de l'usuari, és el menú on podrà interactuar amb el programa de diverses maneres, aquest menú està compost per 4 elements principals:

- 1.1 El primer de tots i el més important és el botó de recomanació. Una vegada clicam aquest botó, ens sortirà una finestra, que ens demanarà els diferents conceptes que volem aprendre d'algun dels cursos de la base de dades.



Imatge 33: Finestra que demana els nostres objectius.

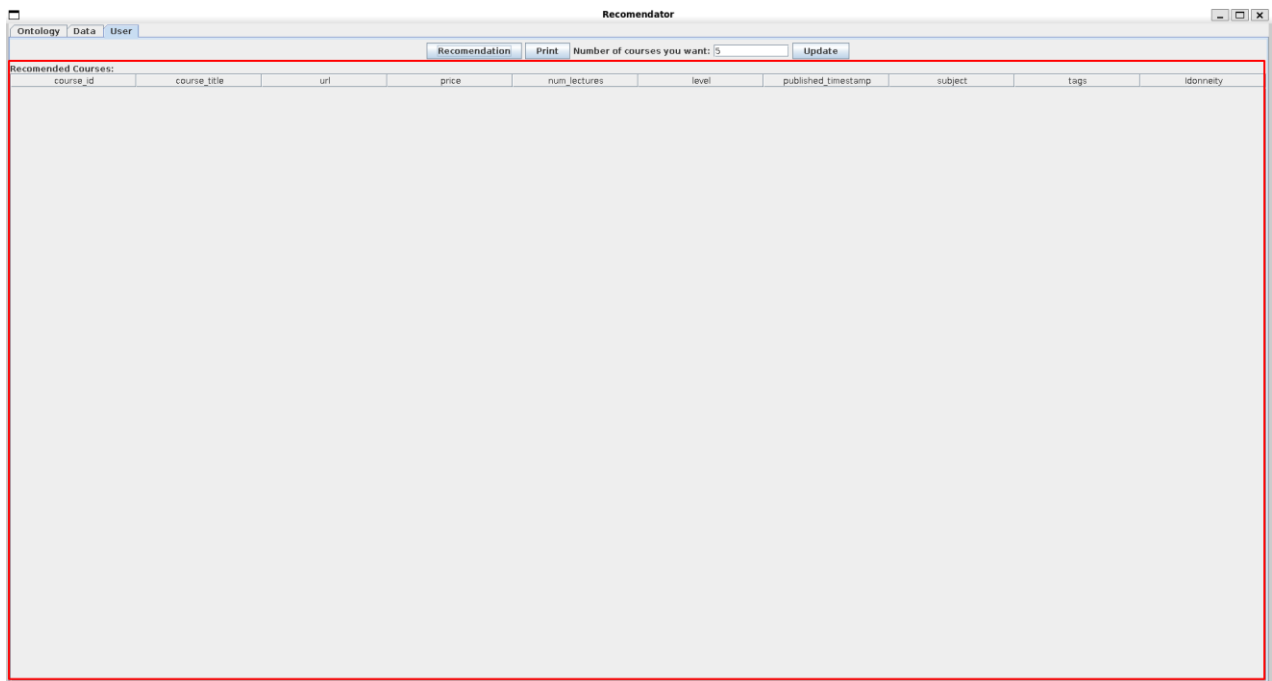
Com podem veure a la imatge anterior, aquesta finestra sortirà un cop haguem clicat el botó de recomanació, en ella podem veure 10 espais en blanc, aquests han de ser rellenats amb els objectius que l'usuari vol assolir, d'aquesta forma el programa els llegirà i realitzarà una recomanació.

Hem establert el nombre en 10, però realment es podria ampliar sense cap problema. De moment, si un usuari té més de 10 conceptes que vol aprendre, haurà de realitzar 2 recomanacions diferents. També cal dir que no es necessari rellenar els 10 espais, si un usuari només té un concepte, basta en posi 1.

S'ha de remarcar que com més objectius doni l'usuari, més acertada serà la recomanació feta pel programa.

- 1.2 El segon element és el botó "print". La funcionalitat d'aquest botó és simplement crear un PDF amb els paràmetres donats per l'usuari, i els cursos que el programa li ha recomanat. D'aquesta forma podem tenir enregistrats els cursos que el programa ha recomanat per si en un futur es vol tornar a consultar la recomanació feta. Més endavant ensenyarem un exemple.
- 1.3 i 1.4 Els dos següents elements van lligats un de l'altre. En primer lloc tenim el 1.3, que consta de un quadre de text, on l'usuari ha d'indicar el número de cursos que vol que se li recomanem, i el 1.4 és el botó per aplicar aquests canvis, el nombre de cursos a visualitzar s'ha establert per defecte en 5. Un cop es clica aquest botó, el programa realitzarà una consulta a la recomanació feta anteriorment i ensenyarà el nombre de cursos que l'usuari consideri, sempre hi quan hi hagi més cursos disponibles en total que el nombre de cursos que l'usuari demana. Cal mencionar que en el quadre de text només en poden insertar caràcters numèrics, i si es dona el cas que volem aplicar canvis però no hi ha cap nombre escrit en el quadre de text, es considerarà que no es volen fer modificacions, per tant la consulta no es farà i no variarà res a l'hora d'ensenyar els cursos.

Aqui acabarem la part superior de la pantalla de l'usuari, ara donem pas a la segona part, que no és interactiva a diferència de la primera. En aquest apartat també explicarem en detall funcions de l'apartat anterior que no hem pogut profunditzar.



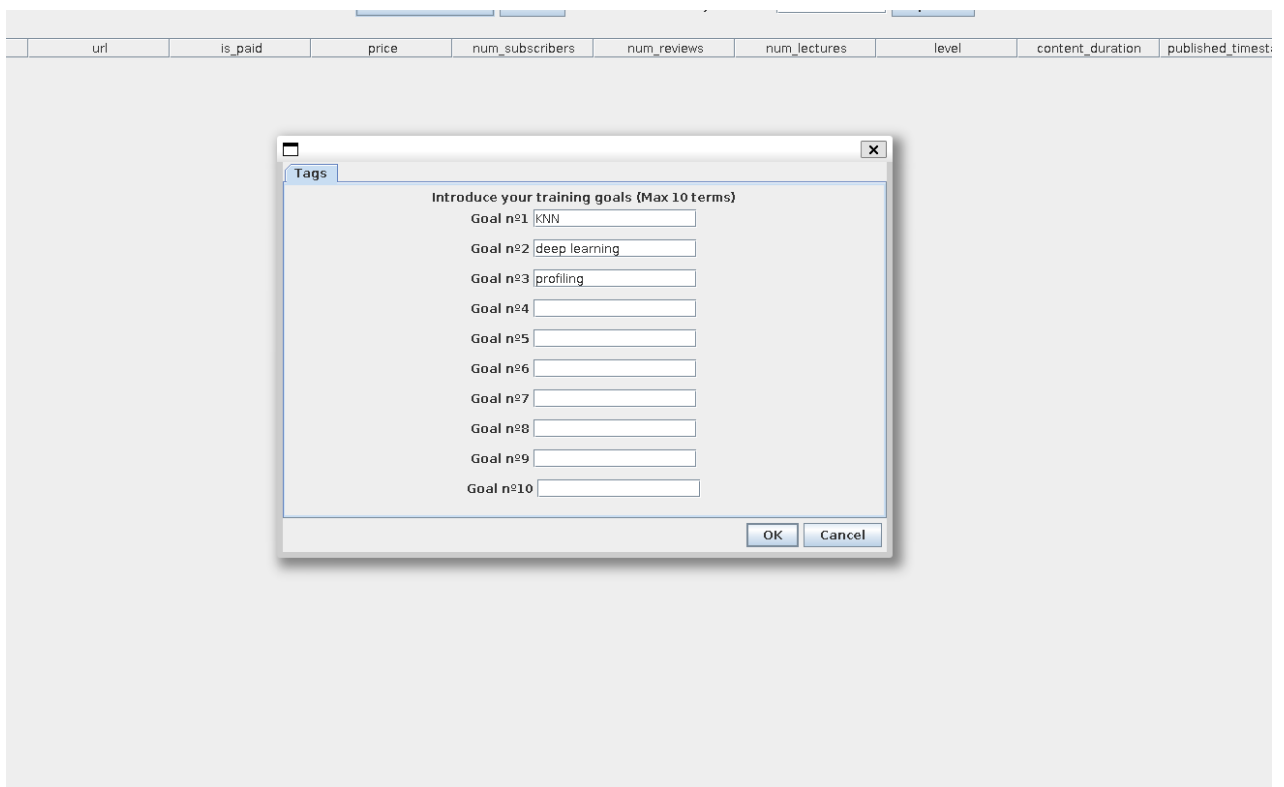
Imatge 34: **Segona part de la pantalla de l'usuari.**

La segona part, la part central i inferior és el panell de visualització dels cursos recomanats. A diferència del panell de la pantalla anterior, en aquest panell només es visualitzaran els cursos que el programa recomani a l'usuari.

Com es pot veure a la imatge anterior, ara mateix el panell està buit ja que no hi ha cap recomanació feta. Cal destacar que les columnes que es visualitzen coincideixen amb el panell de visualització de la pantalla de dades, és a dir, si en la pantalla de dades deixam de visualitzar o afegim visualitzar una columna, els canvis també es veuràn reflectits en el panell de visualització de la pantalla de l'usuari.

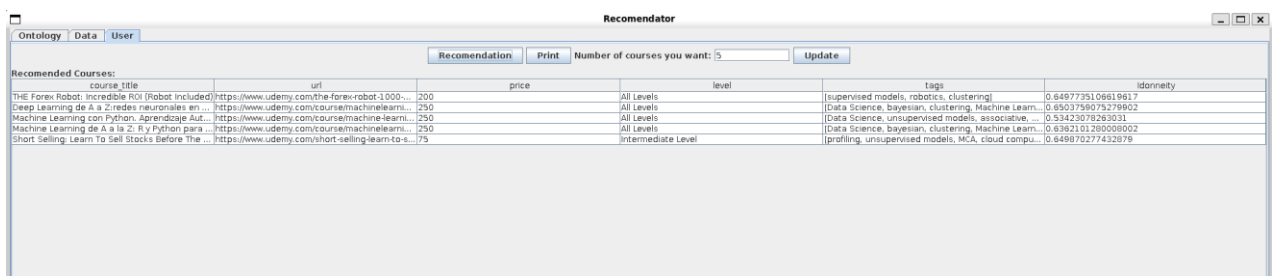
Anem ara a profunditzar un poc més en les funcions que realitzen els botons explicats en l'apartat anterior:

- En la recomanació, realitzarem un exemple del procediment que es realitza, el primer serà introduir els conceptes que es volen aprendre.



Imatge 35: Finestra amb objectius que es volen aprendre.

Una vegada sabem els conceptes que volem aprendre, els introduïm a la finestra que ens sortirà al clicar en el botó de recomanació. Una vegada els hem introduït i cliquem en botó de OK, ens mostrarà els cursos recomanats pel programa.

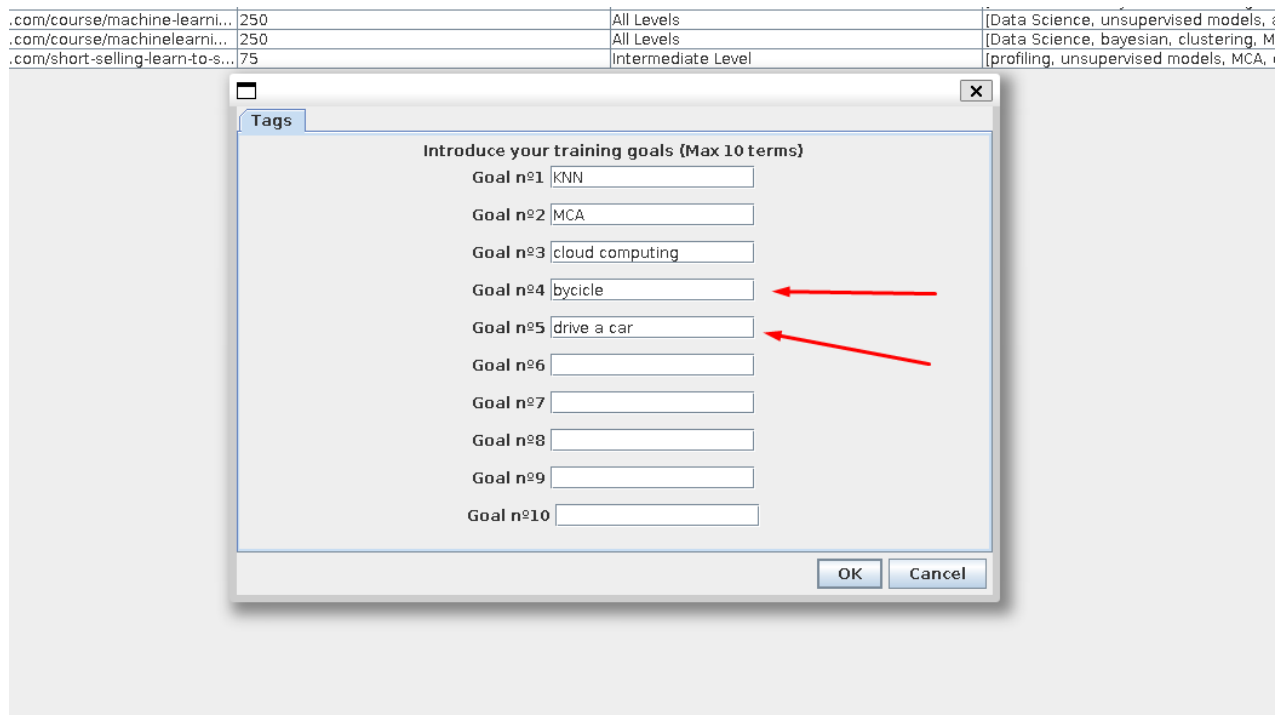


Imatge 36: Cursos recomanats.

Podem veure que el programa ens ha recomanat 5 cursos que considera que podem complir les nostres expectatives amb els objectius que hem facilitat.

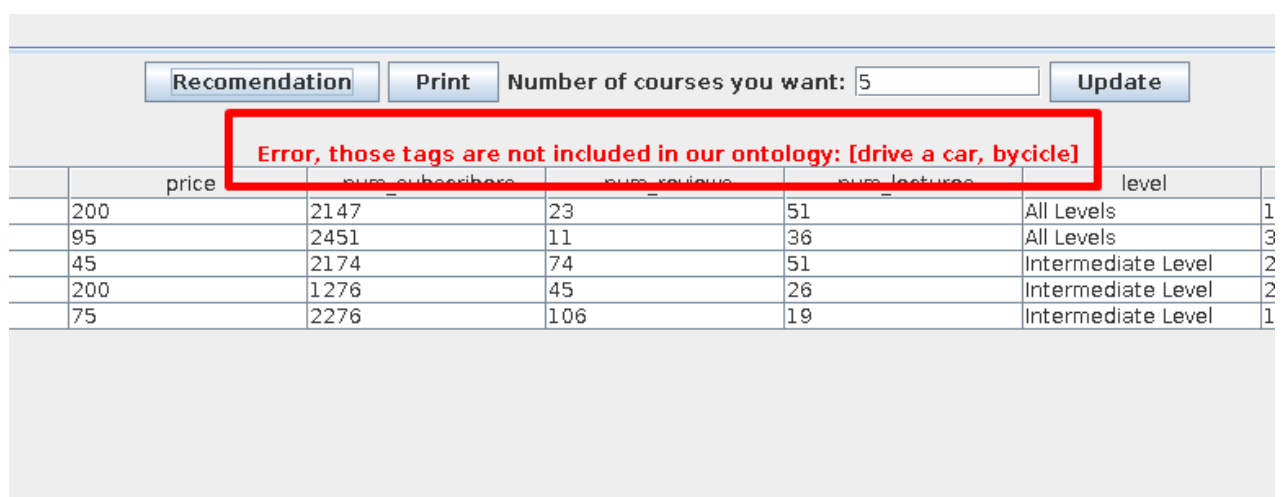
Hi pot haver casos que el nostre programa no detecti o no identifica algun dels objectius que l'usuari doni. En aquests casos es notificarà als usuaris amb un missatge a la part superior de la

pantalla amb els termes que el nostre programa no hagi pogut identificar. Si això arriba a passar, el programa realitzarà una recomanació amb tots els termes que sí s'han pogut identificar.



Imatge 37: Finestra d'objectius amb 2 termes no identificables.

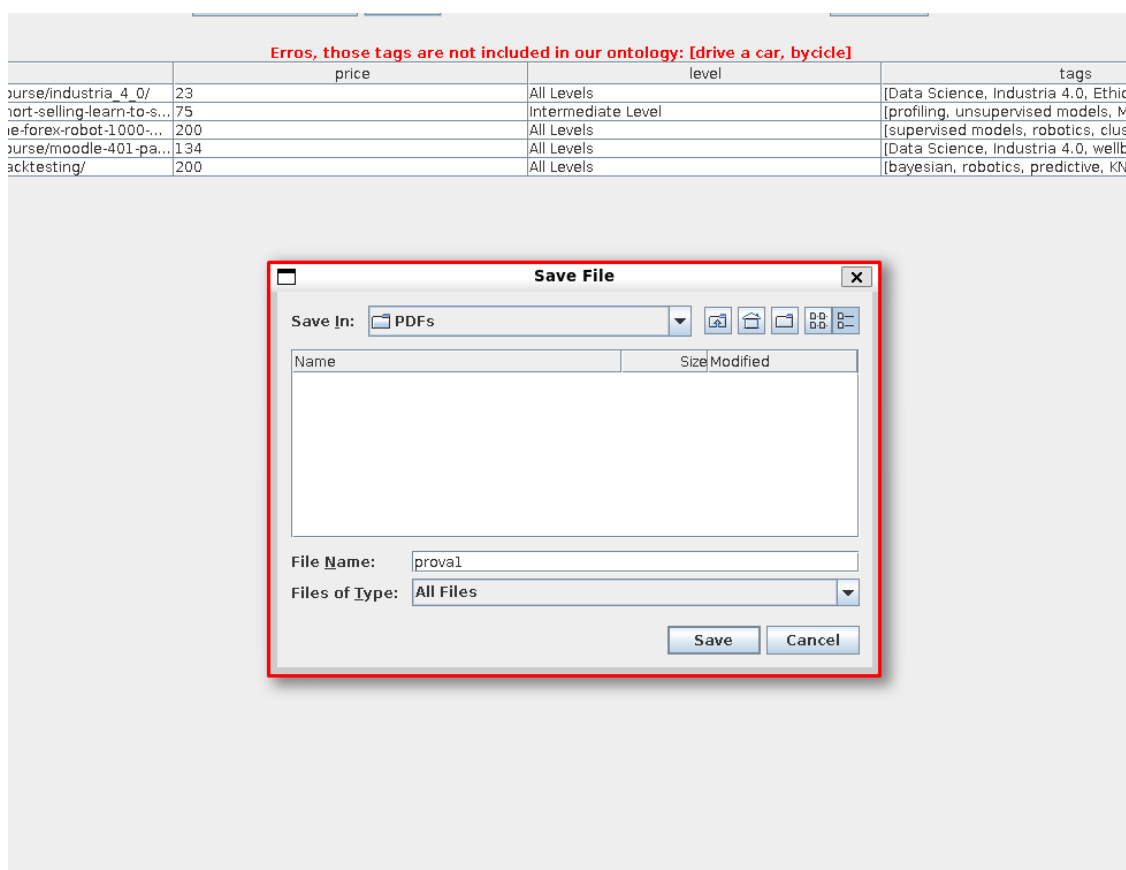
A la imatge anterior hem introduït de nou els termes que volem aprendre, però en aquest cas en les caselles de text 4 i 5 hem introduït termes que estem segurs que el nostre programa no reconeix. En aquest cas, si cliquem el botó de OK, ens apareixerà un missatge a la part superior.



Imatge 38: Missatge d'error de termes no identificats.

Com podem veure en la imatge, el nostre sistema no ha pogut identificar els termes “bicycle” i “drive a car” i per tant ens notifica que els termes no han estat identificats i ha realitzat la recomanació basant-se en els altres 3 termes que havia introduït, que el sistema ha pogut identificar.

- Ara passarem al següent botó que és el de “Print”, com hem mencionat abans, la funcionalitat d’aquest botó és crear un PDF amb els paràmetres que s’han utilitzat en la recerca per realitzar la recomanació. Si cliquem el botó el primer que ens sortirà serà una finestra que ens demanarà on volem guardar l’arxiu i el nom que li volem donar:




Imatge 39: Finestra per guardar el PDF.

Aquí introduïrem el destí de creació del fitxer i el seu nom, en el nostre cas ho decidim guardar en la carpeta PDFs amb el nom de prova1.

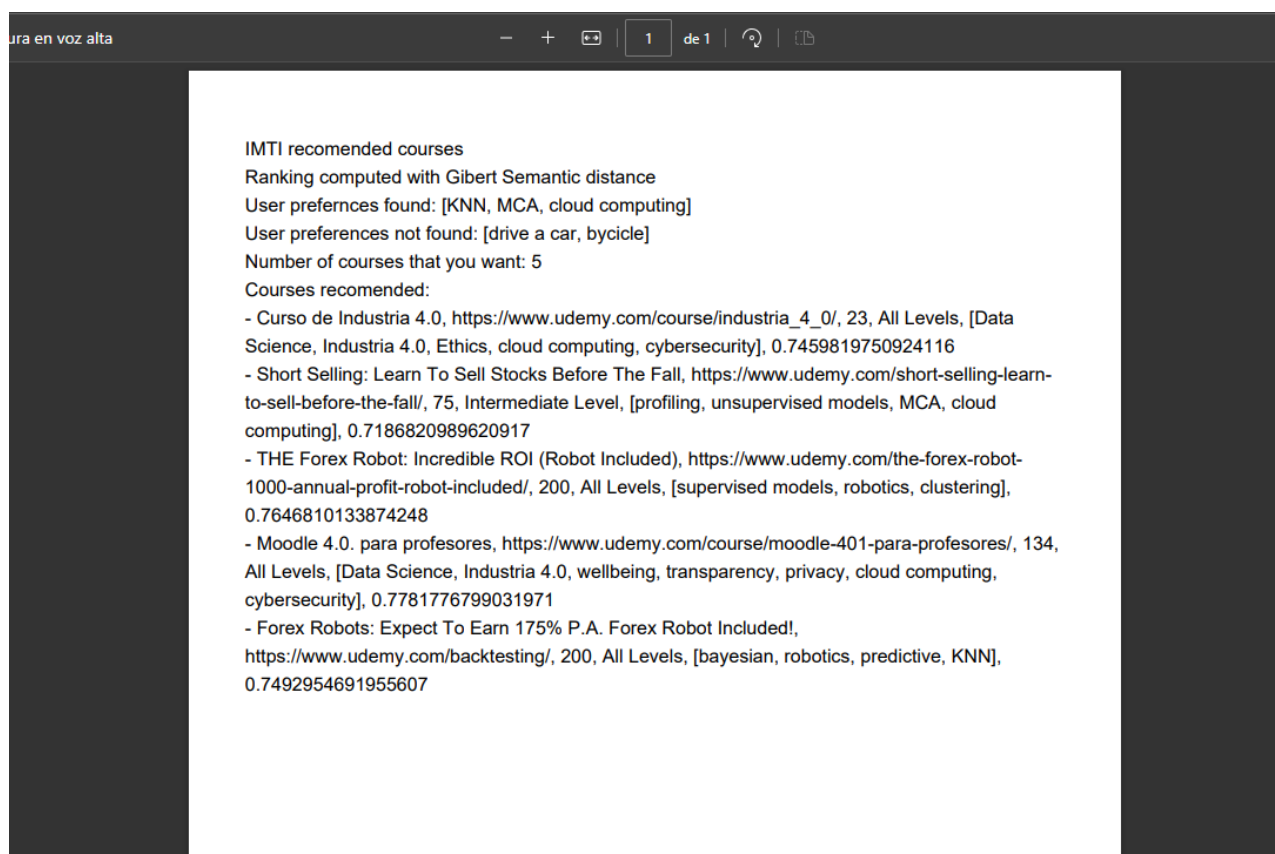
Una vegada clicam guardar, si anam a la carpeta mencionada anteriorment, podem veure com efectivament s’ha creat l’arxiu PDF:

ne > miquel > UNI > TFG-Ideai > PDFs

Nombre	Fecha de modificación	Tipo	Tamaño
 prova1.pdf	16/06/2023 1:25	Microsoft Edge P...	2 KB

Imatge 40: Resultats de la creació del PDF.

En els PDFs es guarda les dades de la consulta feta:



Imatge 41: PDF amb les dades de recomanació guardades.

Com podem veure, les principals dades que guardem en el pdf son:

- Les preferències que s'han pogut identificar, és a dir, són les preferències en les que el programa s'ha basat a l'hora de realitzar la recomanació.

- Les preferències que no s'han identificat, aquestes no s'han tingut en compte a l'hora de processar la recomanació. Pot donar-se el cas que no hi hagi preferències sense identificar.
- El nombre de cursos que l'usuari vol que se li recomanin.
- Finalment tenim les dades dels cursos que el programa ha recomanat, només es mostrarà els valors de les columnes que teníem visualitzades al moment de realitzar la recomanació, d'aquesta forma només es mostrarà la informació més redundat possible, agilitzant la lectura del pdf.

7.2 Emmagatzematge de dades

Una vegada ja hem ensenyat la visualització de la pantalla de dades, passarem a veure parts més concretes de la seva funcionalitat, començant pel seu emmagatzematge, com gestionam aquestes dades i finalment com les representam i interpretam.

7.2.1 Introducció

Per començar sobre les dades, primer realitzarem una petita introducció sobre com guardarem les nostres dades en aquest projecte.

L'emmagatzematge de dades és el procés de guardar i preservar informació de forma estructurada per a ser utilitzada en un futur. Aquest procés resulta necessari per organitzar, accedir i mantenir grans quantitats de dades de forma eficient i fiable.

Avui en dia gràcies als diferents recursos hi ha diferents tecnologies que faciliten l'emmagatzematge de dades, podem destacar:

- [Discs durs](#) [32]: utilitzats en ordinadors i servidors degut a la seva capacitat i el baix cost
- [Internet](#) [33]: mundialment conegut com “almacenamiento en la nube”, és un model que permet accedir a dades a través d'internet en lloc de guardar-los localment.
- [Bases de dades](#) [34]: els sistemes més eficients per organitzar, recuperar i guardar informació de forma eficient. Utilitzades per empreses amb molta quantitat de dades.

Com podem veure hi ha una gran varietat de tecnologies i mètodes per la gestió de dades, en el nostre cas, no ens hem volgut complicar molt i hem optat per guardar les dades facilitades en un simple fitxer .csv en el nostre disc local. Com que durant l'execució del programa la base de dades no variarà i nosaltres no realitzarem cap modificació a cap fila, considerem que emmagatzemar les dades en un disc dur local és la forma més ràpida i eficient.

Una vegada explicat com guardarem les dades dins del nostre ordinador, anirem explicant com el programa guardara aquestes dades.

Per tant per facilitar la situació hem creat classes que ens ajuden a gestionar les dades de un fitxer.

7.2.2 Item

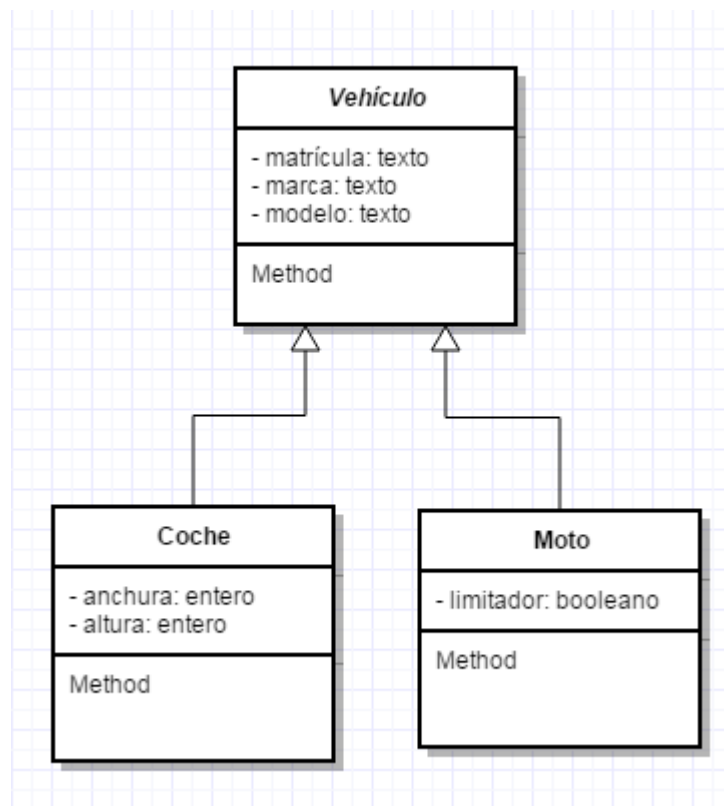
Per començar, havíem de poder representar les files de un fitxer de dades com una instància de una classe creada per nosaltres.

D'aquesta forma varem crear la classe ítem, que té la funcionalitat de representar una fila del fitxer de dades. Per tant, aquesta classe té una sèrie de propietats i funcions que hem considerat necessaris per el correcte funcionament del programa.

Abans de mencionar les propietats que formen aquesta classe, cal destacar la creació d'una altra classe per representar les columnes de cada fila. Degut a que a un mateix fitxer de dades, hi ha diverses columnes que poden tenir diferents tipus de dades, les més destacables són enters, racionals, [booleans](#) [35] i cadenes de caràcters anomenats [strings](#) [36]. Per tant, vam decidir crear una classe anomenada Column. Veguent que totes les columnes comparteixen el mateix atribut, que en el nostre cas seria el valor de la columna com a tal, l'única diferència és el tipus de dades. Per estalviar recursos i aplicar coneixaments adquirits durant el grau, es va optar per definir la classe Column com una [classe abstracta](#) [37], i a partir d'aquesta crear una classe per a cada tipus de dada: ColumnInteger, ColumnBool, ColumnDouble i ColumnString. Totes aquestes classes comparteixen el mètode de `getValue()`, que com el nom indica, retorna el valor de la columna. D'aquesta implementació obtenim diferents avantatges:

- Reutilització de codi: tenim una implementació base comuna a totes les classes, per tant les classes que hereden de la classe abstracta poden aprofitar i reutilitzar el codi, evitant així la duplicació de codi.
- Interfície comuna: una classe abstracta pot definir una interfície comuna per les classes derivades, fent així un comportament comú i que totes les classes derivades compleixin amb característiques o funcionalitats específiques.
- Polimorfisme: les classes abstractes aprofiten el polimorfisme, significat que es pot tractar una instància de una classe derivada com una instància de la classe abstracta, facilitant un codi genèric i flexible.

- Extensibilitat: utilitzant classes abstractes com a base, es pot expandir la seva funcionalitat mitjançant la creació de noves classes derivades, facilitant l'addició de noves característiques i comportaments sense la necessitat de modificar la estructura o el codi ja existent. Exemples d'altres classes abstractes podrien ser una classe per les dates, una pels vectors, etc.



Imatge 42: **Diagrama UML** de la representació d'una **classe abstracta** de vehicles.

Una vegada explicat el mètode que utilitzarem per guardar columnes, tornem a la classe ítem, on ens hem decantat per la senzillesa:

- Hashmap atributs: La forma més òptima de guardar cada atribut de l'ítem és mitjançant un hash, on la clau serà el nom de la columna i el valor serà la instància del tipus de columna.
- Identificador: degut a que treballarem amb un gran nombre de files, hem vist necessari la creació d'un atribut identificador per cada ítem.

Pel que respecte a les funcions, només hem implementat getters, ja que com hem dit abans, el nostre projecte no modificarà les dades que importem.

7.2.3 Item Manager

Una vegada tenim la estructura de un ítem, necessitarem un gestor de instancies d'aquesta classe, la forma més senzilla és amb un manager de ítems, seguint la mateixa estructura que utiliza el Graph, explicat anteriorment.

Aquest funciona de forma molt semblant al graph, quan s'importa un fitxer de dades, aquest gestor va llegint les files i va creant ítems a mesura que acaba la fila, cada ítem amb les seves columnes i els seus valors respectivament. Més endavant entrarem en detall de com es realitza aquesta lectura.

Els atributs del gestor de ítems són bastant predictibles i senzills, els atributs necessaris en un gestor:

- Hashmap ítems: l'estructura indispensable en un gestor com aquest. La clau del hash seria l'identificador de l'ítem i el valor seria la instància del mateix ítem.
- Set cols: a més del hash, es va decidir tenir un set amb els noms de les columnes que hi ha dins del fitxer.

Com es pot veure tant en la classe ítem com en el gestor de ítems només tenim dos atributs que són bastant simples d'interpretar, facilitant així la comprensió i creació de codi.

Òbviament, comptam amb les típiques funcions per obtenir dades sobre un cert ítem.

I també hi ha funcions més específiques per llegir el fitxer de dades i extreure'n la informació per crear els ítems.

7.3 Importació de dades

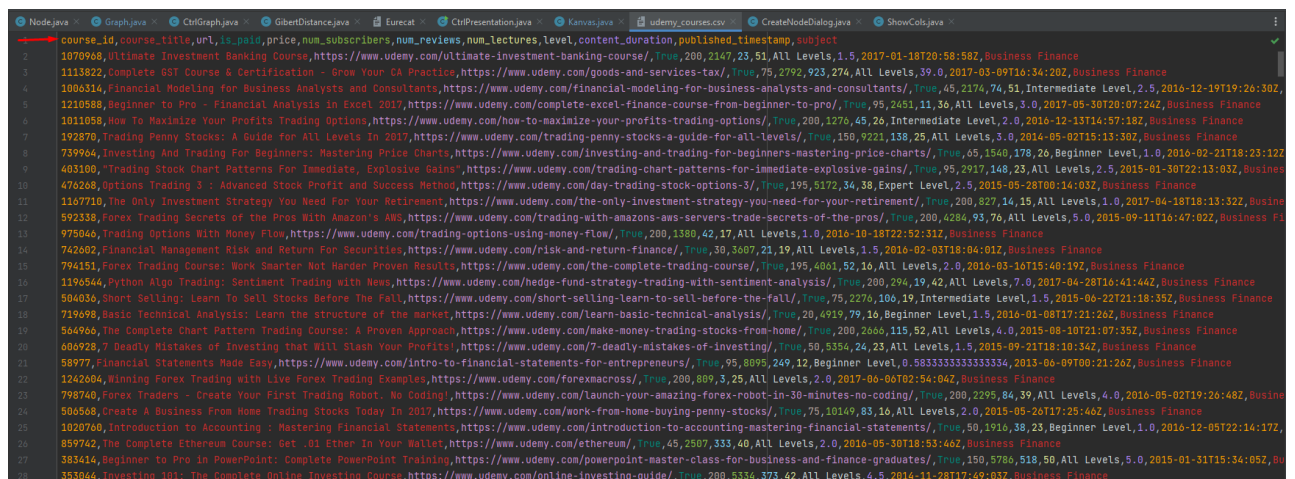
Ara que ja sabem més o menys com funciona la visualització i l'emmagatzematge de dades, explicarem el procés que té el programa per llegir el fitxer de dades i la seva posterior interpretació per crear les instàncies dels ítems de forma correcta amb els valors corresponents de cada columna.

7.3.1 Introducció

Com s'ha explicat anteriorment, el procés d'importació de dades comença quan l'usuari clica el botó d'importació de dades i selecciona el fitxer del qual es volen extreure les dades que conté.

7.3.2 Lectura fitxer

El procés de lectura del fitxer s'inicia quan l'usuari ha elegit quin el fitxer del qual s'extreuen les dades i es guarden dins del programa per després ser visualitzades.



```
1 course_id,course_title,url,is_paid,price,num_subscribers,num_reviews,num_lectures,level,content_duration,published_timestamp,subject
2 1076968,Ultimate Investment Banking Course,https://www.udemy.com/ultimate-investment-banking-course/,True,200,2147,23,51,All Levels,1.5,2017-01-18T20:58:58Z,Business Finance
3 1113822,Complete GST Course & Certification - Grow Your CA Practice,https://www.udemy.com/goods-and-services-tax/,True,75,2792,923,274,All Levels,39.0,2017-03-09T16:34:20Z,Business Finance
4 1086314,Financial Modeling For Business Analysts and Consultants,https://www.udemy.com/financial-modeling-for-business-analysts-and-consultants/,True,45,2174,74,51,Intermediate Level,2.5,2016-12-19T19:26:30Z,
5 1210588,Beginner to Pro - Financial Analysis in Excel 2017,https://www.udemy.com/complete-excel-finance-course-from-beginner-to-pro/,True,95,2451,11,36,All Levels,3.0,2017-05-30T20:07:24Z,Business Finance
6 1011058,How To Maximize Your Profits Trading Options,https://www.udemy.com/how-to-maximize-your-profits-trading-options/,True,200,1276,45,26,Intermediate Level,2.0,2016-12-13T14:57:18Z,Business Finance
7 192878,Trading Penny Stocks: A Guide For All Levels In 2017,https://www.udemy.com/trading-penny-stocks-a-guide-for-all-levels/,True,150,9221,138,25,All Levels,3.0,2014-09-02T15:13:30Z,Business Finance
8 739944,Investing And Trading For Beginners: Mastering Price Charts,https://www.udemy.com/investing-and-trading-for-beginners-mastering-price-charts/,True,65,1540,178,26,Beginner Level,1.0,2016-02-21T18:23:12Z
9 403108,"Trading Stock Chart Patterns For Immediate, Explosive Gains",https://www.udemy.com/trading-chart-patterns-for-immediate-explosive-gains/,True,95,2917,148,23,All Levels,2.5,2015-01-30T22:13:03Z,Business
10 476268,Options Trading 3 : Advanced Stock Profit and Success Method,https://www.udemy.com/day-trading-stock-options-3/,True,195,5172,34,38,Expert Level,2.5,2015-05-28T00:14:03Z,Business Finance
11 1167710,The Only Investment Strategy You Need For Your Retirement,https://www.udemy.com/the-only-investment-strategy-you-need-for-your-retirement/,True,200,827,14,15,All Levels,1.0,2017-04-18T18:13:32Z,Busine
12 592338,Forex Trading Secrets of the Pros With Amazon's AWS,https://www.udemy.com/trading-with-amazons-aws-servers-trade-secrets-of-the-pros/,True,200,4284,93,76,All Levels,5.0,2015-09-11T16:47:02Z,Business FI
13 975046,Trading Options With Money Flow,https://www.udemy.com/trading-options-using-money-flow/,True,200,1380,42,17,All Levels,1.0,2016-10-18T22:52:31Z,Business Finance
14 742602,Financial Management Risk and Return For Securities,https://www.udemy.com/risk-and-return-finance/,True,30,3607,21,19,All Levels,1.5,2016-02-03T18:04:01Z,Business Finance
15 794151,Forex Trading Course: Work Smarter Not Harder Proven Results,https://www.udemy.com/the-complete-trading-course/,True,195,4061,52,16,All Levels,2.0,2016-03-16T15:40:19Z,Business Finance
16 1196544,Python Algo Trading: Sentiment Trading with News,https://www.udemy.com/hedge-fund-strategy-trading-with-sentiment-analysis/,True,200,294,19,42,All Levels,7.0,2017-04-28T16:41:44Z,Business Finance
17 504036,Short Selling: Learn To Sell Stocks Before The Fall,https://www.udemy.com/short-selling-learn-to-sell-before-the-fall/,True,75,2276,106,19,Intermediate Level,1.5,2015-06-22T21:18:35Z,Business Finance
18 719698,Basic Technical Analysis: Learn the structure of the market,https://www.udemy.com/learn-basic-technical-analysis/,True,20,4919,79,16,Beginner Level,1.5,2016-01-08T17:21:26Z,Business Finance
19 564968,The Complete Chart Pattern Trading Course: A Proven Approach,https://www.udemy.com/make-basic-trading-stocks-from-home/,True,200,2666,115,52,All Levels,4.0,2015-08-10T21:07:35Z,Business Finance
20 686928,7 Deadly Mistakes of Investing that Will Slash Your Profits!,https://www.udemy.com/7-deadly-mistakes-of-investing/,True,50,5354,24,23,All Levels,1.5,2015-09-21T18:10:34Z,Business Finance
21 58977,Financial Statements Made Easy,https://www.udemy.com/intro-to-financial-statements-for-entrepreneurs/,True,95,8095,249,12,Beginner Level,0.5833333333333334,2013-06-09T00:21:26Z,Business Finance
22 1242604,Winning Forex Trading with Live Forex Trading Examples,https://www.udemy.com/forextrading/,True,200,809,3,25,All Levels,2.0,2017-06-06T02:54:04Z,Business Finance
23 798740,Forex Traders - Create Your First Trading Robot. No Coding,https://www.udemy.com/launch-your-amazing-forex-robot-in-30-minutes-no-coding/,True,200,2295,84,39,All Levels,4.0,2016-05-02T19:26:48Z,Busine
24 506508,Create A Business From Home Trading Stocks Today In 2017,https://www.udemy.com/work-from-home-buying-penny-stocks/,True,75,10149,83,16,All Levels,2.0,2015-05-26T17:25:46Z,Business Finance
25 1020760,Introduction to Accounting : Mastering Financial Statements,https://www.udemy.com/introduction-to-accounting-mastering-financial-statements/,True,50,1916,38,23,Beginner Level,1.0,2016-12-05T22:14:17Z,
26 859742,The Complete Ethereum Course: Get .01 Ether in Your Wallet,https://www.udemy.com/etherium/,True,45,2507,333,40,All Levels,2.0,2016-05-30T18:53:46Z,Business Finance
27 383414,Beginner to Pro in PowerPoint: Complete PowerPoint Training,https://www.udemy.com/powerpoint-master-class-for-business-and-finance-graduates/,True,150,5786,518,50,All Levels,5.0,2015-01-31T15:34:05Z,Bu
28 353044,Investing 101: The Complete Online Investing Course,https://www.udemy.com/online-investing-guide/,True,200,5334,373,42,All Levels,4.5,2014-11-28T17:49:03Z,Business Finance
```

Imatge 43: Representació de com veu el programa un fitxer .csv.

En la imatge superior podem veure com un llegirà el nostre programa el fitxer csv que volem importar. Hem destacat la primera línia del fitxer, aquesta és la que ens indica el nom de les

columnes que té el fitxer, per tant haurem de tractar la primera línia d'una forma diferent a les altres.

La resta de línies a partir de la primera, si que constitueixen les files originals del fitxer de dades, per tant totes seran tractades de forma similar entre elles.

En el nostre cas, el programa llegeix fitxers de tipus [csv](#)[38], que vol dir “comma separated values”, per tant , cada cop que ens trobem una coma, vol dir que estem canviant de columna i que passem a un nou valor. Això és el que precisament feim en el nostre programa, primer llegim la primera fila i identifiquem les columnes del fitxer i la seva posició, ja que no sempre sabem la posició de les columnes d'un fitxer.

Una vegada ja sabem les columnes i la seva posició procedim a llegir les files.

Com anam llegint caràcter per caràcter, anam guardant el valor de la columna dins un string. Una vegada ens trobem amb una coma, significa que hem canviat de columna, per tant ara toca identificar el tipus de valor que acaban de llegir, ja sigui un enter, un boolean... Una vegada l'hem identificat, el guardam i resetejam el string que utilitzem per a la lectura de caràcters.

Quan s'arriba al final de la línia, el que feim es crear un nou Item amb totes les propietats que havíem llegit i se li assigna un Id al ítem. D'aquesta forma el podem tenir identificat, aquest ítem el guardam al HashMap de ítems.

7.4 Connexió amb l'ontologia

Una vegada ja sabem com es llegeixen i guarden els ítems de la base de dades, anem a veure com es comuniquen la pantalla de dades amb la pantalla de l'ontologia.

La única connexió real que hi ha entre les pantalles de dades i d'ontologia és quan la pantalla de dades demana l'arbre d'ontologies, aquí es realitza una petició a la classe Graph. Mitjançant l'estructura de java [JTree](#)[39], es passa una arrel a la pantalla de dades, un cop rebuda l'arrel, l'arbre s'actualitza i es visualitza a la pantalla de dades.

En la obtenció de l'arrel per part de la classe graph, es realitza de forma recursiva, usant un mètode similar al DFS usat per pintar els nodes, però aplicant recursivitat. Igual que a la visualització, començam pel node arrel, i miram els seus fills i cridam a la mateixa funció per a cada node fill, un cop el fill entra a la funció crea el node per a la classe JTree i a partir d'aquí tenim dos possibles casos:

- Cas base: el node actual no té fills, per tant es retorna el node creat i s'acaba la recursivitat de la funció.
- L'altre cas és que el node sí que té fills i per tant per cada fill es crida a la mateixa funció que ha cridat al pare, creant així la metodologia de recursivitat.

```
1 usage  👤 Miquel Umbert *
public DefaultMutableTreeNode getTree(){

    if(nodes.size()>0){
        return createTree(getNode(getRoot()));
    }
    else return new DefaultMutableTreeNode( userObject: "No data");
}

2 usages  👤 Miquel Umbert
public DefaultMutableTreeNode createTree(Node node){
    DefaultMutableTreeNode root = new DefaultMutableTreeNode(node.getName());
    for(Node child : node.getChildren()){
        root.add(createTree(child));
    }
    return root;
}
```

Imatge 44: Funció que retorna l'arbre de l'ontologia.

D'aquesta forma obtenim l'arbre de l'ontologia a la pantalla de dades, cal dir que si en algun cas és realitzas una petició de l'arbre i no tinguessim cap ontologia creada, no hi hauria cap modificació, ja que la funció detecta que no hi ha cap ontologia i retorna un arbre d'un sol node que indica que no hi ha dades carregades.

7.5 Visualització dels ítems

Passem ara a la visualització dels ítems a la pantalla de dades, que és bastant més simple que la visualització de l'ontologia.

Per la visualització dels ítems hem fet ús de la classe de java [JTable](#)[40], que com hem vist a la pantalla de dades ens permet visualitzar una taula de manera simple i comprensible. Bàsicament l'únic que necessita la classe jTable és el nom de les columnes de la taula i les files amb els atributs.

Pel nom de les columnes la cosa resulta bastant fàcil, ja que com s'ha mencionat abans la classe ItemManager té guardades els noms de les columnes, per tant la seva obtenció es senzilla.

En el cas de les files, la cosa resulta un poc més complexa, com podem interpretar les files de la taula venen a ser els ítems que tenim guardats en el nostre programa.

El que s'ha fet per a que la classe jTable és fer una petició al ítem manager per obtenir les propietats de cada ítem de forma que jTable les accepti per crear una nova fila en la taula.

7.5.1 Ocultar/Mostrar columnes

L'apartat més complicat d'aquesta secció considerem que va ser l'apartat de mostrar i ocultar les columnes que es desitgin i a més que sempre es mantenguin en la mateixa posició.

Per fer això possible, es va crear una propietat per guardar la columna, si aquesta està oculta o no i la posició que ocupa en la taula.

3 usages

```
HashMap<String, Pair<TableColumn, Pair<Boolean, Integer>>> cols;
```

Imatge 45: **Gestió de visualització les columnes.**

Podem veure que es tracta d'un hashMap, amb el nom de la columna com a clau. Després com a valor tenim 3 tipus diferents, el primer és la instància de la columna com a tal, i seguit d'aquesta tenim un boolean que ens indica si la columna esta visible en aquest moment i finalment un enter que és la posició que ocupa la columna dins la taula.

8. Recomanació

Arribem a l'apartat final de l'explicació del projecte, on explicarem tot el procés que és realitza des de que obtenim les dades de l'usuari fins que surt per pantalla els cursos més recomanats per ell.

8.1 Introducció

Per començar explicarem quins models utilitzem a l'hora de fer aquestes recomanacions, on tenim dos models destacables: la distancia semántica de [Gibert](#)[41] i el [KNN](#)[42].

8.1.1 Case based reasoning i K-Nearest Neighbors (KNN)

El raonament basat en casos és un àrea de la intel.ligència artificial basada en el raonament per analogia. Es fonamenta en comparar una escena real amb un històric d'escenes que es troben a la base de casos utilitzant alguna mètrica de referència que permeti ordenar quines escenes s'assemblen més al cas, i assumir que les solucions aplicades per aquestes situacions veïnes

El KNN és un algoritme d'aprenentatge automàtic supervisat usat per a regressió i classificació, és simple però bastant efectiu. És el cas més simple de raonament basat en casos.

Aquest no realitza un procés d'entrenament, sinos que utilitza exemples d'entrenament ja guardats en memòria per prendre decisions. Això el fa bastant útil quan es tenen dades que van canviant amb el temps.

En classificació, el KNN utilitza exemples amb etiquetes ja conegudes per determinar la classe d'un nou exemple, aquest cerca els K exemples més semblants en funció d'una mesura de distància, en el nostre cas la distancia semántica de Gibert, l'elecció del valor K influeix bastant en el rendiment de l'algorisme, ja que una K petita pot estar influenciada pel renou, per l'altre banda, una K massa gran pot suavitzar les diferències entre classes.

En canvi, en regressió, en lloc de determinar una classe, s'utilitza el valor promig o ponderat dels K veïns més propers per predir el valor numèric o continu del nou exemple.

8.1.2 Distància Semàntica de Gibert (superconcept based distance)

Com s'ha mencionat abans, els cursos del nostre repositori tenen molts de tags, i quan l'usuari insereix els seus objectius en forma de llista de preferències, anem a suposar una llista P de preferències $P \{p_1, p_2, \dots\}$, i cada curs té la seva llista de tags, per exemple T1 $\{t_{11}, t_{12}, \dots\}$, T1 fa referència al 1r curs del repositori, i t11 al primer tag del primer curs.

Per tant s'ha de calcular la distància de cada curs respecte a la llista de preferències que ha donat l'usuari.

El que fem primer de tot és obtenir la distància entre dos tags d'aquesta forma:

$$d_{SCD}(c_1, c_2) = \sqrt{\frac{|A(c_1) \cup A(c_2)| - |A(c_1) \cap A(c_2)|}{|A(c_1) \cup A(c_2)|}}$$

Imatge 46: **Càlcul distància entre 2 tags**, referència [paper](#)[41].

El que es fa per saber la distància és obtenir la unió i intersecció dels ancestres dels 2 tags, d'aquests 2 conjunts ens interessa saber el nombre de components de cada un. I a partir de la fórmula de la imatge anterior, és a dir, restem el nombre d'unions menys el d'interseccions, després dividim el resultat entre el nombre d'unions i finalment fem l'arrel quadrada del que ens quedi, així obtenim la distància.

Com hem dit abans els cursos tenen diversos tags i l'usuari pot donar diverses preferències, així que si fem el càlcul de les distàncies entre la llista de preferències P i el curs T1, ens quedarà una matriu de distàncies. El que hem fet és guardar la distància màxima, la mínima i la mitjana de totes les distàncies per cada curs.

Una vegada ho fem per tots els cursos només caldrà ordenar els cursos segons la distància, fent així un ranking de cursos.

Finalment només haurem de retornar el nombre de cursos que l'usuari desitgi de la llista de cursos, visualitzant així els cursos recomanats per pantalla.

8.2 Pre càlcul de distàncies a l'ontologia

Com hem mencionat abans utilitzarem la distància semàntica de Gibert per calcular les distàncies entre els diferents nodes de l'ontologia.

Per dur a terme el càlcul de distàncies i el fet de realitzar la recomanació, s'ha creat una classe anomenada `GibertDistance`, que s'encarregarà del càlcul i de la recomanació.

Per agilitzar el procés, es va optar que una vegada s'importa l'ontologia, les distàncies entre els nodes es calculen per així no haver-los de calcular cada vegada que es realitza una petició de recomanació, d'aquesta forma ens estalviem haver de calcular les distàncies entre tots els nodes moltes vegades.

Un cop l'ontologia està carregada al sistema, l'ítem manager passa els ids dels ítems a la classe `GibertDistance`, i la classe calcula la distància entre els diferents nodes basant-se en la distància semàntica de gibert, utilitzant la propietat dels ancestres de cada node, així tenim les distàncies calculades entre els nodes guardades des d'un principi.

8.3 Obtenció de dades de l'usuari

El primer pas en la recomanació, apart de primer tenir les dades carregades al sistema, és obtenir els objectius que l'usuari vol assolir mitjançant els cursos.

Com s'ha mencionat abans, al clicar el botó de recomanació apareixerà una finestra on l'usuari ha d'introduir els conceptes que vol aprendre. Una vegada s'han introduït, es mandaran a la classe `GibertDistance`, on es distingeixen entre els conceptes que s'han identificat i que per tant estan a l'ontologia i els conceptes que no s'han pogut identificar i que apareixen al missatge d'error que surt a la part superior de la pantalla de l'usuari.

Amb els conceptes que no s'han identificat, simplement es mostren per la pantalla de forma gràfica, en canvi els altres si que són tractats.

8.4 Recomanació de cursos

Una vegada hem obtingut els termes que vol aprendre l'usuari i estan inclosos dins de l'ontologia, el que fem és iterar per tots els cursos del sistema. Per cada curs demanem a l'ítem manager la columna dels tags i per cada tag del curs, el comparem amb tots els objectius que l'usuari ha introduït.

Suposem que un curs te N tags i que l'usuari ha introduït un total de M tags, pues per cada curs calcularem una matriu de $N \times M$, i a cada posició hi haurà la distància semàntica entre els dos conceptes.

Del les matrius, el que feim és guardar la distància mínima, la distància màxima i la mitjana de totes les distàncies, per això hem creat una propietat dins de la classe `GibertDistance` que per cada curs guarda aquestes 3 distàncies relacionades amb la cerca actual per tant les distàncies que queden guardades només són útils per la cerca que ha realitzat l'usuari. Per tant no es tornaran a calcular fins que l'usuari realitzi una nova petició.

8.5 Casos d'ús i proves de concepte

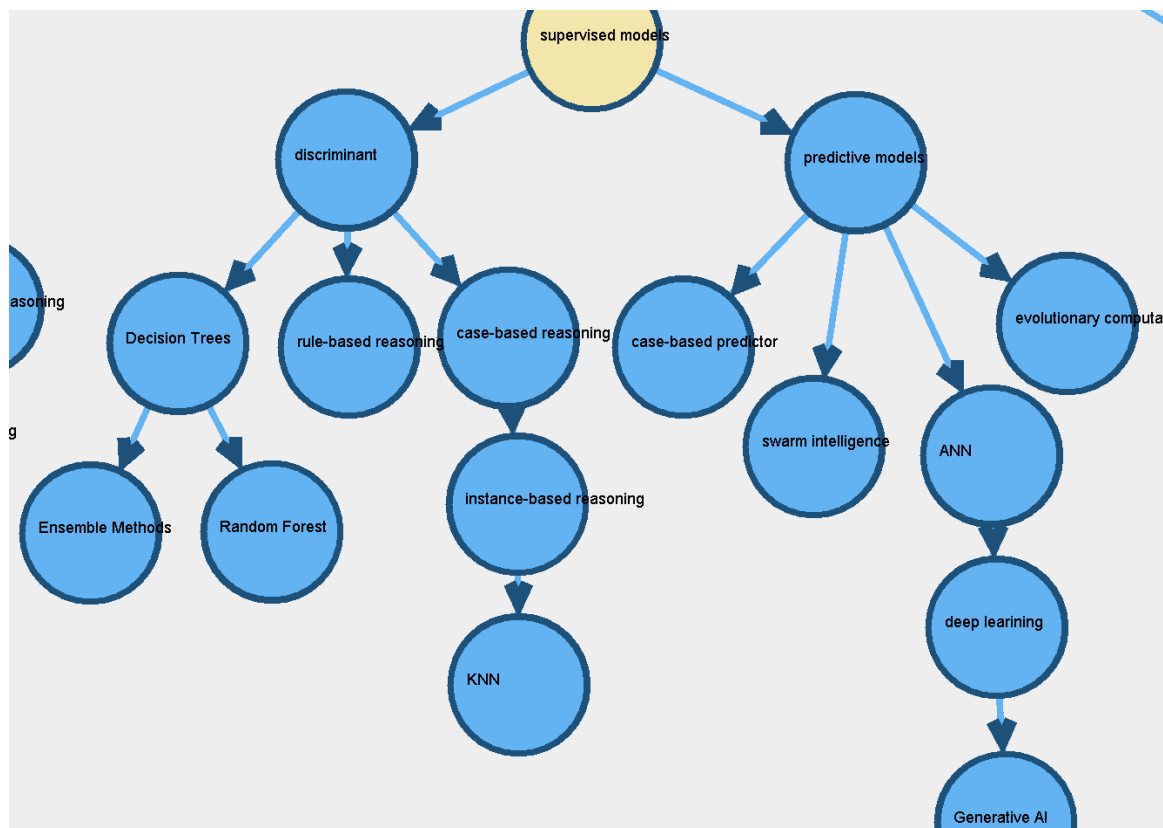
Com s'ha pogut veure, l'objectiu del projecte és construir un recomanador intel·ligent de cursos, i necessitam únicament dues coses:

- L'ontologia sobre la que treballarem.
- El repositori de cursos, d'on extraurem els cursos més adequats per a cada usuari.

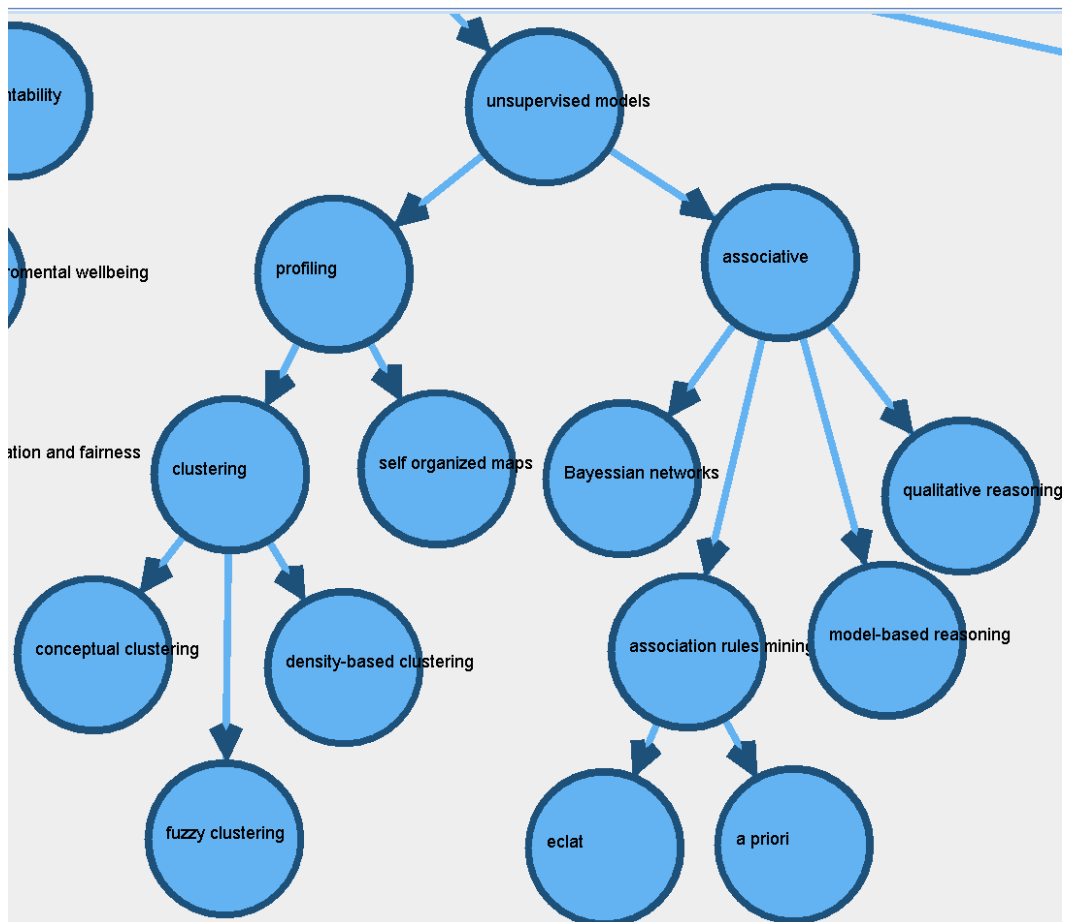
Per realitzar proves de casos d'ús reals, hem creat una ontologia de prova des de 0 amb els següents grans nodes:

- Industria 4.0
- Data Science
- Innovation
- Management

De primeres tenim aquests 4 nodes que donen base a la nostra ontologia, no obstant, per a la realització de proves, s'ha fet un desenvolupament especial de la part de Data Science, ja que la majoria dels cursos de la nostra base de dades estan relacionats en aquest àmbit.



Imatge 47: Fragment de la part de Data Science.



Imatge 48: Fragment de la part de Data Science

Pel que fa al repositori de cursos, vam optar per elegir una base de dades real de Kaggle, en concret vam trobar una gran base de dades de la plataforma de Udemy, una plataforma d'aprenentatge en línia.

Aquesta base de dades la vam adaptar per a que funcionas en el nostre entorn, només va ser necessari afegir la columna de "Tags" i rellena-la amb nodes de l'ontologia perquè el programa fos capaç de realitzar la recomanació.

8.5.1 Casos d'ús

Per a finalitzar amb el projecte s'ha duit a terme una sèrie de casos d'ús per a verificar el funcionament del recomanador intel·ligent.

S'han realitzat 2 consultes com a usuaris de prova per veure el comportament del recomanador:

-En el primer cas tenim un usuari que vol profunditzar els seus coneixements en els camps de : clustering, KNN, bayesian networks.

Un cop hem introduït les dades al recomanador, aquest ens ha donat 5 possibles cursos que poden satisfer la majoria de les necessitats que li hem dit:

course_title
Deep Learning de A a Z: redes neuronales en Python desde cero
Forex Robots: Expect To Earn 175% P.A. Forex Robot Included!
THE Forex Robot: Incredible ROI (Robot Included)
Clustering & Classification With Machine Learning In R
Machine Learning de A a la Z: R y Python para Data Science

Idonneity
0.6383862609866955
0.5301467353925714
0.6164770830454467
0.6963282137137771
0.6378957845061507

tags
[Data Science, machine Learning, density-based clustering, clustering, bayesian networks]
[predictive models, robotics, KNN, bayesian networks]
[supervised models, robotics, clustering]
[Data Science, machine Learning, case-based predictor, clustering]
[Data Science, machine Learning, clustering, bayesian networks]

Imatge 49: Cursos recomanats 1r cas.

En la imatge anterior podem veure els cursos que ens ha recomanat, els seus respectius tags i la distància que s'ha calculat amb la distància semàntica, podem veure que tots els cursos compleixen almenys un dels criteris que s'han enregistrat.

Però de tots els cursos, el que té menys distància és el 2n, com podem veure a la imatge, el 2n curs compleix amb 2 dels requisits que s'han demanat, i a més degut a l'estructura de la nostra ontologia i dels nodes que la componen, el curs que més ens recomana és aquest.

-En el segon cas, elegirem conceptes que no tinguin tanta relació entre ells: Big data, simulation, factorial analysis, improvement.

Un altra cop introduïm les dades i demanam al recomanador 5 cursos:

course_title	Idonneity
Ultimate Investment Banking Course	0.7152883420460937
Beginner to Pro - Financial Analysis in Excel 2017	0.6247546414099726
Curso de Industria 4.0	0.5649140040218702
Moodle 4.0. para profesores	0.6631180270463802
Machine Learning e industria 4.0	0.7340074487797574
Short Selling: Learn To Sell Stocks Before The Fall	0.753009382432691
The Complete Bitcoin Course: Get .001 Bitcoin In Your Wallet	0.7551559467672683
Moodle 4.0. para profesores	0.7169834847324973
Beginner to Pro - Financial Analysis in Excel 2017	0.6889508952677664
Curso de Industria 4.0	0.6452401440803084

tags
[execution, human agency and oversight, Management]
[improvement, cloud computing, Statistics, preparation]
[Data Science, Industria 4.0, cloud computing, ethics, cybersecurity]
[Data Science, Industria 4.0, social and enviromental wellbeing, transparency, privacy, cloud computing, cybersecurity]
[Data Science, machine Learning, Industria 4.0, clustering, cloud computing, bayesian networks]
[multiple correspondence analysis, profiling, unsupervised models, cloud computing]
[Industria 4.0, execution, profiling multivariant, statistical prediction]
[Data Science, Industria 4.0, social and enviromental wellbeing, transparency, privacy, cloud computing, cybersecurity]
[improvement, cloud computing, Statistics, preparation]
[Data Science, Industria 4.0, cloud computing, ethics, cybersecurity]

Imatge 50: Cursos recomanats 2n cas

Com podem veure, a diferència del 1r cas, la distància dels cursos (Idonneity) és major, ja que al demanar conceptes que tenen poca similitud, cap curs els podrà satisfer tots i per tant la seva distància a les preferències de l'usuari serà major.

9. Conclusió del projecte

Per resumir el que hem fet durant aquest projecte, basta dir que hem creat un mentor intel·ligent que recomana itineraris formatius segons uns criteris que dona un usuari. Aquest analitza els criteris i fa una recomanació dels cursos que puguin satisfer les necessitats de cada usuari.

Partim d'una interfície gràfica on es pot agregar o crear una ontologia nova, i que a més es pot modificar dinàmicament a gust de l'usuari. També tenim una pantalla de dades on es carregaran les dades del repositori de cursos que es desitgi, sempre que, de moment, compleixi amb la necessitat de tenir una columna de Tags que formin part de la ontologia.

I finalment tenim la pantalla d'usuari on es realitza la recomanació de cursos mitjançant la introducció d'objectius d'aprenentatge per part de l'usuari. Una vegada introduïts, el programa realitzarà una recomanació dels cursos més adients per aquest usuari.

Durant la realització del projecte he pogut aprendre un gran nombre de coses que n'estic segur, em seran d'utilitat durant tota la vida, de les moltes que hi ha, he destacat les que em pareixen més rellevants:

- Programació d'una interfície visual i interactiva: al principi del projecte tenia pocs coneixements sobre la programació a nivell visual, però amb aquest projecte, he adquirit prou experiència per a construir interfícies més sofisticades.
- Integrar una ontologia dins una base de dades: al principi del projecte no tenia gaire clar com realitzaríem la integració d'una ontologia externa a una base de dades, però durant el projecte, gràcies al suport i mentoria de la directora vaig entendre la forma en què integrariem aquesta ontologia.
- Distància semàntica de gibert: aquest projecte m'ha donat a conèixer aquest mètode de càlcul de distàncies entre les files d'una taula, i l'he trobat de lo més interessant i útil.

Pel que fa al futur del projecte, s'espera que aquest millori en bastants d'aspectes que li falten i que es pugui integrar de forma dinàmica i senzilla a plataformes de cursos online com és el cas del DIH4Cat.

Personalment ha estat un projecte molt interessant i entretingut, que m'ha ensenyat bastantes

coses útils, i no només en el món de la programació, sinó també en el món empresarial, ja sigui establir un pla de treball, fer reunions amb diferents persones i proposar canvis i millores.. Espero que pugui seguir amb la realització d'aquest projecte.

9.1 Línies futures

Encara que s'ha realitzar prou feina en el projecte, encara hi ha diverses coses que és milloraran a mesura que passi el temps, les més destacables que hem mencionat són:

- Generació automàtica de tags de cursos, la idea és que a partir d'una descripció del curs, el recomanador genererà els Tags que vegi convenients.
- Eliminació de les relacions entre els nodes de forma dinàmica, de moment l'única forma d'eliminar relacions és eliminar el node directament

Apart del que acabem de dir, hi ha més coses que no s'han mencionat:

- Al importar una ontologia, el primer node de l'arxiu ".ont" sempre serà l'arrel, així el programa evitarà haver de cercar el node arrel.
- Millora visual del format pdf, ja sigui afegint imatges, centrant textos, adequant les dades de sortida..
- Que el programa recordi les posicions del nodes a la pantalla de visualització, per si l'usuari ha fet modificacions, que els canvis posicionals també es mantenguin.
- Visualització de l'arbre ontològic de la pantalla de dades sigui dinàmic, i no només s'actualitzi quan es pren el botó d'actualització-

A Planificació temporal

A.1 Planificació

Per a que un projecte com aquest tengui exit, s'ha d'elaborar una bona planificació temporal per a que puguem tenir una idea de la seva durada. La UPC estima que un crèdit universitari s'aconsegueix amb una dedicació d'entre 25 i 30 hores. Així que per estimar el temps per completar el projectem si suposem que un crèdit són unes 27 hores de treball i que el treball de fi de grau són 18 crèdits, ens dona que el treball suposarà unes 447 hores de treball.

Podem suposar que la data d'inici del projecte va ser a principis de febrer, i per posar un dia, direm que va ser el 5 de febrer de 2023, ja que l'autor del projecte va començar a cercar informació rellevant, encara que el curs de GEP va iniciar el 20/02/2023. Per una altra banda els torns de lectura comencen el 26 de Juny de 2023, per tant posarem com a data límit 12 de Juny de 2023. Tenint en compte tot això, ens surten uns 127 dies entre la data d'inici i la data límit del projecte. Amb tot això i les hores de dedicació prèviament calculades, ens surt que hauríem de treballar unes 4 hores més o menys per tenir el projecte fet.

Encara així, crec que el més convenient és fixar com a data límit l'1 de Juny de 2023, donant així casi un més de marge per afrontar possibles problemes i imprevists que puguin dificultar l'entrega del projecte. Degut a que ara les hores per dia augmenten i per tant la càrrega de treball, l'autor dedicarà més hores del normal durant els caps de setmana, així el treball que donin les altres assignatures que cursa es repartirà millor al llarg de la setmana. No obstant si l'autor ho considera necessari i oportú, pot dedicar més hores al dia si ho considera.

A.2 Recursos

Com qualsevol projecte, en aquest necessitarem una sèrie de recursos que citarem a continuació.

A.2.1 Recursos Humans

Es necessiten persones per complir certs rols dins del projecte, aquí tenim els membres necessaris:

- Programador (P): A l'hora de crear les classes, models necessaris, un programador es important degut a la complexitat que pot arribar a tenir el codi. Aquest ha de dissenyar l'estructura del programa principal, fer les classes que el componen, realitzar models predictius, tractar possibles errors de codi....
- Escriitor tècnic (ET): Persona encarregada de redactar els documents tècnics relacionats amb el projecte. Apart també pot comentar el codi del programa si el programador l'indica com ho ha de fer. Finalment s'encarregarà de redactar les entregues de GEP i com toca, l'entrega final del projecte.
- Director (D): el director es la persona encarregada de supervisar que el projecte avanci com toca i segons lo que s'ha plantejat. Degut a la seva autoritat aquest pot crear, eliminar o modificar tasques segons com ell vegi.

A.2.2 Recursos Hardware

Degut a la complexitat del projecte, necessitarem un hardware que sigui de certa qualitat per poder fer les tasques:

- MSI GE Raider 63e (OP): Ordenador principal del projecte, on es realitzarà la major part del treball i per tant on invertirem més temps. Aquest té la capacitat i potencia suficient per crear el programa principal i els models de IA sense complicacions, d'una altra forma, tendrem complicacions.
- MSI Modern 14 (OD): Ordenador del director del projecte, sera utilitzar per supervisar el projecte i per validar l'estat de les tasques. Aquest disposa de suficient capacitat computacional per realitzar les tasques corresponents.
- Mouse Corsair (Mou): Mouse utilitzat per l'autor del projecte juntament amb l'ordinador principal. Un recurs prescindible, però que ens pot ajudar a l'hora d'augmentar la velocitat.

- Pantalla AOC (Pan): Pantalla que utilitzarà l'autor per realitzar el projecte, prescindible també però molt útil per visualitzar alhora diferents finestres quan s'utilitzi juntament amb la del ordinador principal.
- Teclat Ozone Tactical (Tec): Teclat que utilitzarà l'autor per tenir major facilitat a l'hora de treballar amb les dues pantalles, aquest facilitarà molt l'escriptura, ja que si es tingués que utilitzar el teclat integrat del ordinador principal, la postura i la comoditat serien dolentes.

A.2.3 Recursos de software

A l'hora de crear el programa, tractar dades, implementar models, comunicar-se... Creiem que seran necessaris els següents recursos de software:

- IntelliJ Idea (In): Programa que permet crear projectes basats en Java.
- Git (G): Sistema de control de versions on la gran majoria de la indústria de la programació utilitza degut a la fiabilitat i efectivitat.
- Trello (Tr): Pàgina web que permet dur la metodologia Kanban, que aquest projecte utilitza.
- Google drive (GD): aplicació web on pots guardar arxius en la red, utilitzat sobretot per intercanvi d'informació entre el director i l'autor.
- Linux Ubuntu (Lin): Sistema operatiu on es dura a terme tot el projecte, degut a la efectivitat d'altres programes, no hi toca haver problema amb la compatibilitat entre Windows i Linux.

Si al projecte al final s'arriba a utilitzar altres llenguatges com R o Python, també necessitarem aquest programari:

- RStudio: aplicació que ens permet tractar les dades del projecte.
- JupyterLab: Programa que permet crear [NoteBooks](#)[43] en Python.

Cal destacar que tot el programari és gratuït, per tant no haurem d'invertir diners per dur a terme el projecte.

A.2.4 Recursos Generals

Apart de tots els altres recursos, també hem de destacar els recursos generals que tot projecte utilitza, ja venen a ser electricitat, l'espai de treball, el temps de cada individu, internet, el mitjà de transport utilitzat per desplaçar-se.

Id	Tasca	Hores	Dependencia	Recursos
GP	Gestió Projectes	77		
GP.1	Contextualització i abast	25	GP.1	D,ET,OP,OD
GP.2	Planificació Temporal	9	GP.1	D,ET,OP,OD
GP.3	Pressupost i sostenibilitat	9	GP.1	D,ET,OP,OD
GP.4	Document Final	19	GP.1,GP.2,GP.3	D,ET,OP,OD
GP.5	Reunions	15		P,D
TP	Treball Previ	18		P,OP
D	Disseny de l'eina	5	TP	P,OP
CRD	Creació repositori dades	2	TP	P,OP
VEO	Visualització i edició Ontologies	75	TP	
VEO.1	Organització i documentació	20		P,OP
VEO.2	Programa i finestra de visualització	5	VEO.1	P,OP,In,G,Lin
VEO.3	Funcions finestra	50	VEO.2	P,OP,In,G,Lin
CO	Creació ontologia	80	VEO	
CO.1	Organització i documentació	20		P,OP
CO.2	Creació de les classes	30	CO.1	P,OP,In,G,Lin
CO.3	Models	60	CO.2	P,OP,G,Lin
U	Usuari	50	TP	
U.1	Identificació vistes i tipus	2		P,OP,In,G,Lin
U.1	Creació classes	10	U.1	P,OP,In,G,Lin
U.2	Guia visualització usuari	40	U.2	P,OP,In,G,Lin
PREC	Procés de recomanació	30		P,OP
IA	Interpretació i anàlisis	20	CO.3	
IA.1	Proves de concepte	10		P,OP,D
IA.2	Interpretació resultats	10		P,OP,D
DOC	Documentació	75		P,OP
DOC.1	Recollida Informació	25		P,OP,GD
DOC.2	Document final	50		P,OP,GD
PR	Presentació	15	DOC	
Total		447		

Taula 1: Resum de les tasques del projecte

A.3 Planificació estimada i real

A.3.1 Gestió del risc

A l'abast del projecte, vam explicar els possibles obstacles que presentava el nostre projecte, i per això hem creat una sèrie de possibles solucions als problemes que vam dir:

- **Falta d'experiència:** aquest problema té una solució bastant trivial. Només haurem de dedicar hores a aprendre explícitament els elements necessaris per dur a terme el projecte. Com es pot observar, la majoria de tasques, tenen una subtasca anomenada *organització i documentació de la tasca*, on es dedicara una gran part del temps a aprendre el que sigui necessari per complir aquella tasca. També cal dir que em fixat la data límit a un parell de setmanes abans de la data límit real, en cas que necessitéssim un temps extra. Podríem dir que té un alt risc de probabilitat.
- **Obtenció de les dades:** l'obtenció de les dades anirà destinada a la tasca de *Models*, en el nostre cas només podem esperar a que DIH4Cat ens faciliti els seus cursos disponibles per a que nosaltres poguem fer les proves i models. En el pitjor dels casos que no es poguessin facilitar les dades d'hora, nosaltres crearíem exemples reals per poder seguir amb el treball. No creiem que ens hagi de preocupar molt, així que el risc és baix.
- **Mala informació de l'usuari:** una solució trivial també és aplicable a aquest problema, per poder obtenir les dades de forma correcta, bastarà amb afegir una sèrie de condicions per a que es pugui realitzar la recomanació, si les condicions no es compleixen, saltarà un missatge d'error que informarà a l'usuari dels paràmetres que estan incorrectes. Aquestes condicions poden ser camps obligatoris, limitació del tipus de dades que es poden introduir,... Aquest risc s'ha de tenir en compte sempre, però intentarem que la seva aparició sigui mínima.
- **Quantitat de temps:** amb aquest problema, la única solució viable es anar al dia amb el treball i dur una bona organització de les altres assignatures també, per a que no es junti tot el treball en un període curt de temps, pareix una solució fàcil, però a vegades organitzar-se correctament, pot ser la tasca més complicada en un projecte. Aquest risc s'ha de tenir en consideració, té alta probabilitat que passi però l'impacte no seria molt gran.

Cal remarcar que els obstacles no suposen una gran pèrdua d'hores, ja que en la planificació de les tasques ja s'han tingut en compte la majoria d'ells. A més que tenim les setmanes extres per si acas algun dels obstacles es complicas més de l'esperat.

B. Pressupost i sostenibilitat

B.1 Identificació dels costos

B.1.1 Costos humans

Com s'ha mencionat en la planificació temporal, per realitzar les tasques, són necessaris recursos humans, de hardware i finalment recursos de software. Lo més important a l'hora de realitzar les tasques, són els recursos humans, per això hem detallat les persones necessàries per realitzar les tasques:

- **Programador:** persona que s'encarrega de programar tot el codi, realitzar els models corresponents, investigar sobre els problemes que poden sorgir i la forma de tractar-los. Aquest serà el que participarà en més tasques, que són: treball previ (**TP**), visualització i edició ontologies (**VEO**), creació ontologia (**CO**), usuari (**U**) i la interpretació i anàlisi (**IA**).
- **Escriptor tècnic:** persona amb la tasca d'escriure la documentació i els documents tècnics. Aquest participara sobretot en les tasques de gestió de projectes (**GP**) i la de documentació (**DOC**)
- **Director:** la persona encarregada de supervisar el projecte i d'assegurar-se que tot va al dia i com es va plantejar, aquesta persona participa en la tasca de gestió de projectes (**GP**) i en la interpretació i anàlisi (**IA**).

Ara sabent quina tasca correspon a cada individu, definirem una taula per detallar el cost de contractació de cadascuna d'aquestes persones.

Rol	Salari anual	Preu per hora	Assignació
Director projecte	46.200	26,03€	Director
Programador	28.500	16,05€	Autor
Escriptor tècnic	20.670	11,64€	Autor

Taula 3: Salari anual de cada treballador del projecte. Es té en compte que es treballa a jornada completa unes 1775, fent la mitja de lo que diu la pàgina [asesorias](#)[44], i la pàgina [jobted](#)[45], ens diu els salaris anuals mitjans de cada professió.

B.1.2 Costos de software

En el nostre projecte, els recursos de software són de codi lliure, per tant són totalment gratuïts, encara que en alguns productes, hi ha versions que si es poden pagar per obtenir funcions addicionals. En el nostre cas, les funcions bàsiques són més que suficients per assolir el projecte, per tant el nostre cost total de software es 0. No obstant, inclourem els recursos usats en la taula final dels costos per tenir una taula més detallada i precisa dels costos totals del projecte.

B.1.3 Costos de hardware

Els recursos de hardware en l'àmbit informàtic tenen un grau elevat de cost , degut a la seva potencia i a altres factors com els costos de producció, costos de transport, impostos aplicats al nostre país... Cal dir que per dur a terme el projecte, no són necessaris uns dispositius tan cars com els de la taula que teniu a continuació, peor justament són els que disposaven l'autor i el director del projecte:

Dispositiu	Preu	Any de fabricació
MSI Modern 14	800€	2020
MSI GE Raider 63e	1200€	2018
Mouse corsair	45€	2019
Pantalla AOC	100€	2020
Teclat ozone tactical	60€	2020

Taula 4: Costos dels dispositius hardware i el seu any de fabricació

Hem de detallar que els costos són de quan es van obtenir els dispositius, l seu cost po variar de l'actual i del preu de llançament de cada producte.

B.1.4 Estimació costos del projecte

Finalment en aquest apartat, donarem l'estimació del pressupost del nostre projecte en forma de taula, així queda més llegible i fàcil d'entendre. Detallarem els costos i més tard donarem una explicació de com hem obtingut els resultats

Costos per tasca

Tasca	Detalls	Cost
GP.1- Contextualització i abast	25 hores-> Director, escritor tècnic	941,75€
GP.2- Planificació temporal	9 hores-> Director, escritor tècnic	339,03€
GP.3- Pressupost i sostenibilitat	9 hores-> Director, escritor tècnic	339,03€
GP.4- Document final	19 hores-> Director, escritor tècnic	715,73€
GP.5- Reunions	15 hores-> Director, escritor tècnic	565,05€
TP- Treball Previ	18 hores-> Programador	288,90€
D- Disseny de l'eina	5 hores -> Programador	80,25€
CRD- Creació repositori	2 hores -> Programador	32,10€
VEO.1- Organització tasca	20 hores-> Programador	321,00€
VEO.2- Programa i finestra	5 hores-> Programador	80,25€
VEO.3- Funcions finestra	50 hores-> Programador	609,90€
CO.1- Organització tasca	20 hores-> Programador	321,00€
CO.2- Creació classes	30 hores-> Programador	481,50€
CO.3- Models	30 hores-> Programador	481,50€
U.1- Identificació vistes	2 hores-> Programador	32,10€
U.2- Creació classes	10 hores-> Programador	160,50€
U.3- Guia visualització usuari	38 hores-> Programador	642,00€
PREC	30 hores -> Programador	481,50€
IA.1- Proves de concepte	10 hores-> Programador, Director	420,80€
IA.2- Interpretació resultats	10 hores-> Programador, Director	420,80€
DOC.1- Recolida Informació	25 hores-> Escritor tècnic	291,00€
DOC.2- Document final	50 hores-> Escritor tècnic	582,00€
PR- Presentació	15 hores-> Escritor tècnic	174,60€
Costos per les tasques	447 hores de treball	11.662,79€

Taula 5: Costos de les tasques, la taula mostra el nom de la tasca, les hores invertides i el treballador involucrat i finalment el cost de cada tasca.

Costos generals

	Espai de treball	1.906,90€
Centre de treball	570€/ mes de mitjana, 447 hores	1722,52€
Internet	40€/mes, 447 hores	120,87€
Electricitat	0.2133 €/kWh hora punta	3,51€
T-Jove, T-Usual	Necessaries per desplaçar-se 4 mesos	60€
	Hardware	148,79€
MSI GE 63	447 hores, preu: 1200€	117,78€
MSI Modern 14	62 hores, preu: 800€	10,89€
Corsair Mouse	447 hores, preu: 45€	4,41€
Pantalla AOC	447 hores, preu: 100€	9,81€
Teclat Ozone	447 hores, preu: 60€	5,90€
	Software	0,00€
Google Drive	Sense cost	0,00€
IntelliJ	Sense cost	0,00€
Git	Sense cost	0,00€
Trello	Sense cost	0,00€
Linux Ubuntu	Sense cost	0,00€
Total costos generals	Espai treball + hardware + software	2.055,69€

Taula 6: Costos generals del projecte, dels que destaquem els costos de l'espai de treball, de hardware i de software.

Pressupost general del projecte

Cost inicial projecte	Costos per tasca + generals	13.718,48€
Cost adicional per contingencia	Marge del 15%	2057,78€
Cost total	Cost inicial + contingencia	15.776,26€
Cost per incidencia		2.834,3€

Cost final del projecte	Cost total + per incidencia	18.610,56€
-------------------------	-----------------------------	-------------------

Taula 7: Pressupost final del projecte.

Explicació pressupostos

En aquest apartat, explicarem i donarem a entendre com hem tret els costos de cada taula.

En la primera taula, la de **costos per tasca**, hem considerat que la millor forma de realitzar els càlculs és assignar els recursos humans dels que disposam a cada tasca i un cop estan totes assignades, utilitzar la taula 1 dels preus per hora. Ara que ja sabem les hores estimades de cada tasca i el preu dels professionals per hora, es pot calcular el preu de cadascuna simplement multiplicant el nombre d'hores, per la suma dels costos de cada professional assignat a ella.

Ex: **Tasca I**, **H_i** -> nombre hores de la tasca i, **T_{ij}** -> preu per hores del treballador j assignat a la tasca i, **N_i** -> nombre de treballadors assignats a cada tasca I

$$\text{Preu tasca I} = H_i * (T_{i0} + .. + T_{i(n-1)})$$

Una vegada ja sabem la fórmula per calcular el cost de les tasques, només és aplicar la fórmula a cada tasca i sumar el cost de cada tasca per obtenir el cost total de les tasques, en el nostre cas serà de 11.662,79€.

Ara que ja tenim els costos per tasca, procedirem a calcular els costos generals, que se divideixen en espai de treball, hardware i software.

En el cas del software, com hem mencionat abans, és fàcil de calcular, ja que tot el nostre software ha tingut un cost de 0€, ja que s'ha utilitzat programari lliure en aquest projecte.

Pels recursos de hardware, hem calculat l'amortització de cada producte, per calcular-la, hem usat la següent fórmula:

$$\text{amortització} = \text{cost_producte} / (\text{anys_de_vida} * 4 * 253)$$

A la fórmula, hem usat els valors 253 pels dies laborals que hi ha, i 4 pel nombre d'hores diàries de treball que realitzariem. A més s'ha rebuscat la vida útil dels productes de hardware i hem fet una estimació que ens dona ens uns 4,5 anys de vida útil entre tots els productes. Un cop traiem l'amortització, només l'hauré de multiplicar pel nombre d'hores que hem usat el producte.

Finalment, a l'apartat d'espai de treball, hem inclòit diversos recursos. Pel desplaçament a l'oficina de treball usarem només transport públic durant els 4 mesos que dura el projecte, per aquest desplaçament, usarem una T-Jove, que end dura 3 mesos i té un cost de 40€ i una T-Usual pel mes restant que ens queda, amb un cost de 20€. Basant-nos en la pàgina web de venta i alquiler de immobiliari [Idealista](#)[46], avui en día una oficina d'uns 50m², costa aproximadament unes 570€ al mes. Per calcular el cost total de espai en el temps que dura el projecte, farem l'operació de: ((570€/mes * 12 mesos) * 447 hores de treball) / 1775 hores laborables = 1.722,52€.

Apart de l'espai de treball en sí, aquest també consumeix uns recursos que van fora de l'alquiler mensual, el cost actual de l'electricitat s'estima en 0,2133 €/kWh en hora punta. Mirant les especificacions dels ordenadors i del monitor del projecte, ens surt uns 31,4W amb un total de

524 hores de treball. Fent els calculs, estimam que gastarem uns 16,45kWh. Així que amb l'operació de $16,45\text{kWh} * 0.2133 \text{ €/kWh} = 3.51\text{€}$ en electricitat.

També ens fixarem en la tarifa d'internet, amb la velocitat que ofereix, el preu mitjà d'aquest tipus de tarifes és de 45€/mes. Fent ús de la mateixa fórmula que el cost de l'espai de treball: $((40\text{€/mes} * 12 \text{ mesos}) * 447 \text{ hores de treball}) / 1775 \text{ hores laborables} = 120,87\text{€}$ d'internet. Fent les sumes dels costos generals, ens dona un total de 2.055,69€.

Ara que ja tenim calculats els 2 principals costos del projecte, obtenim un cost inicial de 13.718,48€, però hem de tenir en compte que durant el projecte poden sorgir imprevistos i fallos que poden conduir a un retràs en l'entrega del projecte. Hem posat com a remei un sobrecost del 15% mes o menys per fer front a aquests possibles imprevistos que poden sorgir. Afegint aquest sobrecost, el preu del projecte s'eleva fins els 15.776,26€.

Apart de possibles incidències, podem afegir un cost adicional per si hi ha retrasos a l'hora d'acabar el projecte. Segons el grau del problema que sorgesqui, afegirem un capital, com més gran sigui el seu grau, més capital s'ha d'afegir. Això és només un possible exemple del que pot passar. Per exemple, degut a la falta d'experiència creiem que hi ha un 50% que tardem més de l'esperat per acabar el projecte, deiem que tardam 3 setmanes més del compte per fer el projecte, per tant són 60 hores més de feina, així que s'han de sumar a cada treballador involucrat, del cost de les hores, només agafarem el 50% respectiu.

També cal considerar la perdua de dades, però en el nostre cas és una externa a nosaltres per tant hem de suposar que el nostre proporcionador sempre les tindrà de forma correcta.

Pot passar que és produesqui una pèrdua o dany de material, per exemple que sens rompi l'ordinador, donat que és poc probable, li assignarem un 5% de probabilitat, el seu cost seria de 60€, un 5% del cost de l'ordinador del programador.

Finalment suposar una pèrdua de progrés, degut a que el controlador de versions és bastant fiable, li donarem un risc del 10% tirant a l'alta. Per calcular el cost d'aquest risc, agafarem el cost total de les tasques i traurem un 10%, en el nostre cas són 1.162,7€.

Tenint en compte les incidències que hem mencionat, tenim un gast adicional de: 2.834,3€.

Ho traiem de la formula: $0,5 * 60 \text{ hores} * (26,03 + 16,05 + 11,64) + 0,1 * 11.662,79 + 0,05 * 1200$. Afegint tot el cost, ens surt un preu final absolut de 18.610,56€.

B.2 Control d'estimacions

En els apartats anteriors hem estimat uns costos possibles, però durant el projecte potser que ens desviem dels valors estimats anteriorment. Per això calcularem la desviació de cada tipus de tarea per poder fer rectificacions amb altres tasques. Establirem indicadors per saber la desviació de cada tasca.

El primer a tenir en compte són els recursos humans. La situació pot passar per diversos motius, molta càrrega i fatiga de treball, treballar en un camp nou desconegut.. Per calcular la seva desviació hem usat la fórmula:

$$\text{Desviació recurs humà} = \text{Total hores consumides} * (\text{Preu estimat} - \text{Preu real})$$

En lo que respecta al software, no podem tenir cap desviació ja que no tenim cap cost. La cosa és diferent en hardware que sí pot haver una desviació. La desviació és basa en l'ús real que li donarem.

$$\text{Desviació amortització} = \text{Preu per hora} * (\text{hores d'ús estimat} - \text{hores d'ús real})$$

Ara ens fixarem en els costos generals. L'únic que consideram que pot tenir una desviació significativa és l'ús de l'electricitat, ens basarem en una fórmula molt semblant a la del hardware:

$$\text{Desviació electricitat} = \text{Preu per hora} * (\text{hores d'ús estimat} - \text{hores d'ús real})$$

Finalment farem l'estimació dels costos per un possible retràs. Si el projecte acaba en el temps estimat, ens podrem ahorrar aquests costos i hi haurà una desviació significativa:

$$\text{Desviació costos retràs} = \text{Cost estimat} - \text{cost real}$$

B.3 Informe de sostenibilitat

B.3.1 Opinió personal

Cada cop que és parla sobre sostenibilitat, pens amb un parell de conceptes però els 3 que destaquen més són l'aspecte ambiental, social i econòmic. Són els que la majoria de gent te a consideració a l'hora d'aprovar un projecte, com afectarà al medi ambient, si la societat farà un canvi dràstic, si el govern i/o la gent s'ho podrà permetre... Ara donarem, un parell de detalls sobre l'impacte que pot tenir el nostre projecte en relació amb la sostenibilitat:

- **Ambiental:** el nostre projecte no té un impacte directe en el medi ambient, però alomillor el pot tenir de forma indirecta, si el nostre recomanador és efectiu i eficient, pot fer que la gent gastí menys electricitat (usant l'ordinador per cercar un itinerari que li pugui agradar), i menys combustible (no haver-se de desplaçar a centres de recomanació d'itineraris). Però aquest aspecte és en el que menys influeix el nostre projecte.
- **Social:** en aquest crec que és on el projecte té menys impacte, el seu èxit pot provocar un canvi a l'hora de decantar-se per un itinerari o per un curs diferent. Pot impulsar a que més gent estudiï el que li agrada o fins i tot gent que vol aprendre nou coneixement. Pot fer més eficient la vida dels ciutadans, si no saben a que és volen dedicar o decantar en la seva vida laboral, el mentor pot solucionar-ho de forma ràpida i precisa.
- **Econòmic:** l'aspecte econòmic ja l'hem estudiat a l'apartat anterior tots els seus costos i possibles riscos que poden sorgir del projecte. Hem fet una estimació del pressupost que costarà el projecte, i per tant podem entendre l'impacte que pot tenir econòmicament. Si ens fixem en els ciutadans, serà un programa a l'abast de tothom i que el seu únic consum seria el gast de l'electricitat consumida durant el seu ús.

En els següents apartats, contestarem a algunes preguntes sobre la sostenibilitat del nostre projecte.

B.3.2 Aspecte ambiental

Heu estimat l'impacte ambiental del projecte?

Degut a que el projecte es basa en un software de recomanacions, aquest no tindria cap impacte visual ni auditiu respecte al medi ambient. Però com hem dit abans, indirectament pot ajudar a reduir la quantitat d'energia elèctrica produïda i del la benzina consumida.

Heu plantejat alguna estratègia per a que el projecte gastí menys recursos i tengui un menor impacte ambiental?

Com que el món de la programació és molt extens, hi ha la possibilitat de reutilitzar codi de projectes semblants a n'aquest per tal de reduir el nombre d'hores usant els recursos materials, menys hores usant el hardware, menys electricitat consumida. A més intentarem aplicar tècniques bàsiques per estalviar electricitat: aprofitar tot el que puguem la llum natural, si ens podem desplaçar caminant o transport públic millor, només enchufar els aparells electrònics quan sigui necessari.

Com es resol actualment el problema, de quina forma millorarà ambientalment la teva solució?

Actualment hi ha centres i webs on segons les teves preferències et donen un possible llistat d'itineraris que poden encaixar amb el teu perfil. Nosaltres volem aportar una solució més eficient respecte a les webs actuals, i en front dels centres, la gent interessada ja no s'hauria de desplaçar fins ells, cosa que pot reduir la contaminació ambiental.

B.3.3 Aspecte social

Que creus que t'aportarà a nivell personal aquest projecte?

Serà un repte personal per a mi, ja que mai que he estat involucrat en un projecte d'aquestes dimensions, però sobretot crec que el que més m'aportarà serà coneixement, ja sigui programar una interfície des de 0 en java, models d'intel·ligència artificial i tractament de dades, i això a part de satisfacció personal, pot donar molt en el futur laboral. A més l'autor es planteja si fer el màster d'intel·ligència artificial, i aquest projecte pot decidir si finalment es decanta per fer-ho o no.

Com es resol actualment el problema, de quina forma millorarà socialment la teva solució?

Com hem dit la gent ja no s'hauria de desplaçar fins als centres on rebrien una recomanació, ni haurien de passar tant de temps en altres llocs menys eficients que el nostre.

Existeix una necessitat real del projecte?

Sí, considerant que molta gent no sap els possibles cursos/ itineraris que hi ha disponibles actualment, la nostra eina els ajudaria a decidir-se pel que més li agradi, coneixent tots els que tenen a l'abast.

B.3.4 Impacte econòmic

Has estimat el cost del projecte?

Sí, en apartats anteriors s'ha fet una estimació del cost total del projecte, tenint en compte tots els recursos necessaris per fer-ho, recursos humans, de hardware i de software; els costos generals que tindrà el projecte, les instal·lacions, el desplaçament; i fins i tot s'ha tingut en compte els possibles gastos que poden tenir imprevistos i retrasos dins del projecte.

Com es resol actualment el problema, de quina forma millorarà econòmicament la teva solució?

La gent ja no s'hauria de desplaçar, cosa que ajudaria a ahorrar en combustible, ni hauria d'estar molt de temps per webs cercant, per tant també resulta en un estalvi energètic. Però el que més estalviaran serà temps, que podran dedicar a altres coses que no sigui la recerca d'itineraris.

Bibliografia

- [1] -> IDEAI: <https://ideai.upc.edu/en> 20/03/2023
- [2] -> Digital Innovation Hub de Catalunya: <https://dih4cat.cat/es/> 19/03/2023
- [3] -> Pagina web de Kaggle: <https://www.kaggle.com/> 18/06/2023
- [4] -> Microsoft inverteix 10.000 milions en IA: <https://rb.gy/dk8deq> 13/03/2023
- [5] -> Apple compra 25 companyies de IA: <https://rb.gy/mmxptx> 13/03/2023
- [6] -> Google inverteix 500 milions en rival de chat GPT: <https://rb.gy/jdfewc> 13/03/2023
- [7] -> Facebook compra empresa de IA per 1000 milions: <https://rb.gy/k5sqgo> 13/03/2023
- [8] -> Chat GPT en l'educació: <https://rb.gy/22vago> 13/03/2023
- [9] -> Camps on la IA ha avançat: <https://rb.gy/grff4v> 13/03/2023
- [10] -> Chat GPT4: <https://openai.com/product/gpt-4> 20/03/2023
- [11] -> Recomanador de cursos UDEKA: <https://udeka.es/> 19/03/2023
- [12] -> Llei de protecció de dades de la UE: <https://shorturl.at/cCIKM> 17/05/2023
- [13] -> Comissió Europea sobre la intel·ligència artificial: <https://shorturl.at/moNSV> 17/05/2023
- [14] -> Metodologia Kanban: <https://es.wikipedia.org/wiki/Kanban> 19/03/2023
- [15] -> Pàgina principal de Trello: <https://rb.gy/wypq9t> 20/03/2023
- [16] -> Documentació Java: <https://docs.oracle.com/en/java/> 20/03/2023
- [17] -> Llenguatge R: <https://www.r-project.org/> 20/03/2023
- [18] -> Llenguatge Python: <https://www.python.org/> 20/03/2023
- [19] -> Pàgina principal de IntelliJ: <https://www.jetbrains.com/es-es/idea/> 20/03/2023
- [20] -> Pàgina principal de JetBrains: <https://www.jetbrains.com/es-es/> 20/03/2023
- [21] -> Explicació de Git: <https://es.wikipedia.org/wiki/Git> 20/03/2023
- [22] -> Definició d'ontologia: <https://shorturl.at/bcHMN> 08/06/2023
- [23] -> Patró disseny singleton: <https://refactoring.guru/es/design-patterns/singleton> 08/06/2023
- [24] -> Pàgina de Udemy: <https://www.udemy.com/> 08/06/2023
- [25] -> Definició de radioButtons: <https://shorturl.at/fgkHQ> 08/06/2023
- [26] -> Estructura de dades en arbre: <https://estructurasite.wordpress.com/arbol/> 08/06/2023
- [27] -> Definició DFS: <https://shorturl.at/xABQV> 18/06/2023
- [28] -> Explicació recursivitat: <https://es.wikipedia.org/wiki/Recursi%C3%B3n> 08/06/2023
- [29] -> Classe canvas: <https://shorturl.at/esux6> 18/06/2023
- [30] -> Classe pila: <https://shorturl.at/bxHOQ> 18/06/2023
- [31] -> Definició de checkbox: <https://victomanolo.wordpress.com/checkbox-en-java/> 08/06/2023
- [32] -> Definició discs durs: https://es.wikipedia.org/wiki/Unidad_de_disco_duro 08/06/2023
- [33] -> Definició de la cloud: https://es.wikipedia.org/wiki/Almacenamiento_en_nube 08/06/2023
- [34] -> Definició base de dades: <https://shorturl.at/elQR6> 08/06/2023

- [35] -> Definió de booleans: <https://shorturl.at/noqR3> 08/06/2023
- [36] -> Definió strings: <https://developer.mozilla.org/es/docs/Glossary/String> 08/06/2023
- [37] -> Definió classe abstracta: <https://shorturl.at/pJOVX> 08/06/2023
- [38] -> Explicació fitxer CSV: <https://shorturl.at/evBCD> 08/06/2023
- [39] -> Classe JTree: <https://shorturl.at/cxJQ8> 08/06/2023
- [40] -> Classe JTable: https://t.ly/b7y_ 08/06/2023
- [41] ->Distancia Gibert: <https://shorturl.at/nFGR8> 08/06/2023
- [42] ->Explicació KNN: <https://shorturl.at/irxyY> 08/06/2023
- [43] ->Explicació JupyterNotebook: <https://rb.gy/tct3fb> 20/03/2023
- [44] -> Pàgina d'assessoria: <https://rb.gy/shpigu> 10/03/2023
- [45] -> Pàgina de JobTed: <https://www.jobted.es/salario> 10/03/2023
- [46] -> Idealista: <https://www.idealista.com/> 11/03/2023