

# PROJECTES DE PROGRAMACIÓ

## QP1920

### GRUP 11

CARLES CAPILLA CÁNOVAS  
ALBERT COMA COMA  
PERE GRAU MOLINA  
MUHAMMAD MERAJ ARSHAD

# Índex de continguts

<b>Relació de classes</b>	<b>3</b>
<b>Descripció funcionalitats</b>	<b>4</b>
Entrada de dades	4
Preprocés de dades	4
Exportació de dades preprocessades	4
Importació de dades preprocessades	4
Aplicació de l'algorisme	4
Consulta i modificació de regles	4
Exportació de regles d'associació	5
Validació de regles	5
<b>Estructures de dades i algorismes utilitzats</b>	<b>5</b>
HashSet<Element>	5
HashMap<key,value>	5
LinkedHashMap<key,value>	6
<b>Diagrama de casos d'ús</b>	<b>7</b>
<b>Descripció dels casos d'ús del sistema</b>	<b>8</b>
1. Cas d'ús: Entrada de dades	8
2. Cas d'ús: Definició d'atributs	9
3. Cas d'ús: Assignar valors per defecte al preprocés (opcional)	9
4. Cas d'ús: Entrada del format de preprocés	10
5. Cas d'ús: Preprocés de dades	11
6. Cas d'ús: Exportació dades preprocessades	11
7. Cas d'ús: Importació dades preprocessades	12
8. Cas d'ús: Generar regles d'associació	12
9. Cas d'ús: Consulta de regles d'associació	13
10. Cas d'ús: Eliminació de regles d'associació (opcional)	13
11. Cas d'ús: Modificació de regles d'associació (opcional)	14
12. Cas d'ús: Exportació de regles d'associació	14
13. Cas d'ús: Importació de regles d'associació	15
14. Cas d'ús: Validació de regles d'associació	15
15. Cas d'ús: Conversió de regles d'associació (opcional)	16
<b>Diagrama de classes de l'especificació del sistema</b>	<b>17</b>

# Relació de classes

La relació de classes implementades per a cada membre del grup és la següent:

## **Carles Capilla Cánovas:**

- CjtAtributs
- Atribut i els seus fills (AtributBool, AtributInt, AtributString)
- Interval
- CtrlAtributs
- Tots els drivers associats a les classes anteriors

## **Albert Coma Coma:**

- ReglaAssociació
- CtrlRegles
- Tots els drivers associats a les classes anteriors

## **Pere Grau Molina:**

- AtributProcessat
- Registre
- CjtRegistres
- CtrlRegistres
- Tots els drivers associats a les classes anteriors

## **Muhammad Meraj Arshad:**

- CjtReglesAssociació
- CtrlRegles
- Tots els drivers associats a les classes anteriors

# Descripció funcionalitats

## Entrada de dades

Importació de les dades al nostre sistema per tal de poder realitzar un posterior anàlisi. L'entrada de dades en importar un fitxer amb un determinat format. Aquest format consta d'una primera fila amb el nom dels tots els atributs separat per coma. A continuació, en els següents línies (registres) hi haurà el valor de cada atribut. Els atributs que no tinguin valor s'indiquen amb NULL.

## Preprocés de dades

Per al preprocés de dades hem de tractar només les dades que posteriorment necessitem analitzar:

- Descartar les dades que no són necessàries.
- Discretitzar les dades que ho necessitin (assegurar-nos que totes les dades són binàries).
- Definir intervals pels atributs que ho necessitin (atributs numèrics).
- Aplicar els valors per defecte als atributs no presents (si cal)

## Exportació de dades preprocessades

Les dades un cop preprocessades han de poder ser exportades a algun format extern, ja sigui un fitxer simple en format txt, json, etc.

## Importació de dades preprocessades

Poder importar un fitxer que ha estat prèviament exportat del nostre programa. Les dades que s'importen des d'aquest mètode ja estan preprocessades i llestes per aplicar-li l'algorisme.

## Aplicació de l'algorisme

Un cop tenim les dades preprocessades, procedim a aplicar l'algorisme per tal de generar les regles d'associació.

Cada regla té associada un valor per a la confiança i un altre per al suport (?).

## Consulta i modificació de regles

Donat un conjunt de regles, poder consultar-les totes en una interfície intuïtiva i senzilla. Més endavant, es podria afegir la possibilitat de modificar i eliminar regles.

## Exportació de regles d'associació

Donat un conjunt de regles d'associació obtingut a partir de l'aplicació del nostre algorisme, poder extreure aquestes regles a un fitxer en format binari,txt,etc

## Importació de regles d'associació

Donat un conjunt de regles d'associació obtingudes prèviament i exportades a un fitxer del tipus desitjat, importar-les a la nostra interfície per tal de poder validar-les o aplicar-les per a un nou conjunt de dades.

## Validació de regles

Donat un conjunt de regles d'associació i un nou conjunt de dades, validar les regles amb el nou conjunt de dades per comprovar la seva correctesa (comparar els valors de suport i confiança obtinguts).

## Estructures de dades i algorismes utilitzats

### HashSet<Element>

Aquesta estructura de dades s'ha utilitzat per implementar llistes tant de valors com d'altres classes per tal d'evitar elements duplicats i, per tant, emmagatzemar elements en llistes d'un menor tamany i, per conseqüència, més fàcils i ràpides de recórrer.

- HashSet<String> : Emprat en la definició d'una llista de valors possibles d'un atribut de tipus String.

### HashMap<key,value>

Aquesta estructura de dades s'ha utilitzat per poder emmagatzemar un conjunt de parelles de valors.

- HashMap<String,String> : Emprat per poder consultar dins el driver del controlador d'atributs la llista dels nom i tipus d'atributs definits a partir del fitxer i l'especificació de l'usuari.
- HashMap<String,ArrayList<String>> Emprat per poder consultar dins el driver del controlador d'atributs la llista dels atributs amb:
  - En el cas d'un atribut de tipus String s'emmagatzema el nom de l'atribut i la llista de valors possibles que pot tenir.

- En el cas d'un atribut de tipus Float s'emmagatzema el nom de l'atribut i una llista de la definició de cada interval assignat a l'atribut.
- En el cas d'un atribut de tipus Boolean s'emmagatzema el nom de l'atribut així com una llista amb els dos valors possibles que pot prendre.

## LinkedHashMap<key,value>

Aquesta estructura de dades s'ha utilitzat per implementar un conjunt de parells de valors de manera que es conservi en tot moment l'ordre en què s'han inserit aquests.

- LinkedHashMap<LinkedHashSet<AtributProcessat>,Integer>: Emprat per guardar tots els itemsets amb la seva freqüència al generar regles d'associació.

## LinkedHashSet<Element>

Aquesta estructura de dades s'ha utilitzat per implementar un conjunt d'elements d'un tipus de manera que es conservi en tot moment l'ordre en què s'han inserit aquests i sense tenir elements repetits.

- ArrayList<LinkedHashSet<AtributProcessat>>: Emprat per guardar una llista de candidats de mida 'k' per generar les regles d'associació.

# Diagrama de casos d'ús



(Es pot trobar aquesta imatge al directori **DOCS** de l'entrega).

# Descripció dels casos d'ús del sistema

## 1. Cas d'ús: Entrada de dades

- **Nom:** Entrada de dades en format taula

- **Actor:** Usuari

- **Comportament:**

- 1) L'usuari indica al sistema que vol entrar dades.

- 2) El sistema demana el fitxer de dades.

- 3) L'usuari puja el fitxer de dades amb el següent format:

La primera línia del fitxer conté el nom de tots els atributs, separats per coma. Després a partir de la segona línia cada atribut té el seu valor, també separat per coma. Els atributs que no tenen valor en un registre s'ha de posar NULL. Per tant, cada fila de la columna correspon a una entrada diferent per a cada atribut.

Un exemple seria el següent:

- atr1, atr2, atr3,..., atrN
- valor1, valor2,NULL, ..., valorN;
- valor1, NULL, valor3, ..., valorN;

- 4) El sistema comprova el format del fitxer.

- 5) El sistema comprova que no hi ha valors NULLs

- 6) Fi del cas d'ús.

- **Errors possibles i cursos alternatius:**

- 4.1a) El sistema retorna una excepció indicant "El fitxer adjuntat no té un format correcte."

- 4.1b) Es torna a repetir el pas 3 del cas d'ús.

- 4.2a) El sistema troba un valor null d'un atribut.

- 4.2b) El sistema pregunta si vol posar un valor per defecte per l'atribut de forma interactiva.

- 4.2c) L'usuari respon amb "SÍ" o "NO".

- 4.2d) Si l'usuari respon SÍ:

- 4.2d.1) El sistema demana el valor per l'atribut.

- 4.2d.2) L'usuari escriu el valor per l'atribut.

- 4.2d.3) El sistema guarda el valor de l'atribut.

- 4.2e) Si l'usuari respon NO, aleshores el sistema continua treballant amb l'atribut amb valor NULL.

- 4.3a) El sistema troba un valor null d'un atribut. (opcional)



4.3b) El sistema pregunta si vol posar un valor per defecte per l'atribut a partir de les dades d'aquest atribut.

4.3c) L'usuari respon "sí" o "no".

4.3d) El sistema posa un valor per defecte a l'atribut a partir de les dades de l'atribut.

4.3e) Segueix el cas d'ús.

5a) El sistema demana a l'usuari valors per defecte pels atributs que contenen NULLs.

## 2. Cas d'ús: Definició d'atributs

- **Nom:** Definició d'atributs

- **Actor:** Usuari

- **Precondició:** Les dades han d'haver estat entrades prèviament.

- **Comportament:**

- 1) El sistema mostra un llistat de tots els atributs que apareixen a les dades entrades amb una primera suposició del tipus que pot ser cada atribut segons les dades entrades.

- 2) El sistema demana a l'usuari que indiqui els tipus de cada atribut. Els possibles tipus són:

- a) **Binari:** el valor és true (1) o false (0).

- b) **Categòric:** el valor és una string. En aquest cas, l'usuari ha d'indicar tots els possibles valors que pot prendre l'atribut (enum).

- c) **Numèric:** el valor és un número (float).

- 3) L'usuari indica el tipus de cada atribut

- 4) El sistema comprova que els tipus concordin amb les dades entrades

- a) En el cas d'haver dades sense valor (NULL):

- i) El sistema demana a l'usuari valors per defecte pels atributs que contenen valors NULLs

- ii) L'usuari introdueix els valors per defecte

- 5) Fi cas d'ús.

- **Errors possibles i cursos alternatius:**

- 4) El sistema retorna una excepció indicant que les dades entrades no corresponen amb el tipus definit de l'atribut.

## 3. Cas d'ús: Assignar valors per defecte al preprocés (opcional)

- **Nom:** Assignar valors per defecte al preprocés

- **Actor:** Usuari

- **Precondició:** L'usuari ha indicat el preprocés a seguir.

- **Comportament:**

- 1) El sistema comprova si queden atributs amb preprocés sense definir.
- 2) Si no en queden saltar al pas 5.
- 3) Si en queden es crea una discretització segons el seu tipus:
  - a) Si es tracta d'un bool no es realitza cap preprocés ja que la binarització ve implícita.
  - b) Si es tracta de String s'assumeixen tots els valors de l'atribut com possibles
  - c) Si es tracta d'un float, per defecte es binaritzarà fent la mitjana de tots els valors presos per aquell atribut i considerant els intervals de (X sent la mitjana) valors < X i X >= valors.
- 4) Si en queden es torna al pas 3.
- 5) Fi cas d'ús.

## 4. Cas d'ús: Entrada del format de preprocés

- **Nom:** Entrada del format de preprocés

- **Actor:** Usuari

- **Precondició:** Les dades han d'haver estat entrades prèviament.

- **Comportament:**

- 6) L'usuari informa al sistema que vol entrar el format del preprocés de dades.
- 7) El sistema demana un fitxer de dades.
- 8) L'usuari puja el fitxer al sistema amb el següent format:
  - a) Hi haurà una línia per atribut.
  - b) Per referenciar el valor de l'atribut, s'utilitzarà el valor especial {val}
  - c) Si l'atribut es vol discretitzar, el format serà (format i exemple):  
`nomAtr1=valor1:cond1; valor2:cond2; ...; valorN:condN;`  
`temperatura=alt:{val}>30; mitjà:{val}<=30 && {val}>20;`  
`baix: {val}<=20`
  - d) Si l'atribut es vol binaritzar, el format serà (format i exemple):  
`nomAtr2=condició`  
`alcoholEnSang={val}<0.25`
- 9) El sistema comprova que el format del fitxer sigui vàlid.
- 10) Fi cas d'ús.

- **Errors possibles i cursos alternatius:**

- 4a) El sistema retorna una excepció indicant que el format del fitxer no és vàlid.

## 5. Cas d'ús: Preprocés de dades

- **Nom:** Preprocés de dades
- **Actor:** Usuari
- **Precondició:** Les dades han d'haver estat entrades
- **Comportament:**
  - 1) L'usuari informa al sistema que vol preprocessar les dades.
  - 2) El sistema demana si l'usuari ho vol fer interactiu o amb un fitxer
  - 3) El sistema recorre totes les dades i les converteix segons el format especificat.
  - 4) Fi cas d'ús.
- **Cursos alternatius:**
  - 2a) Si l'usuari vol fer interactivament , aleshores l'usuari entra les dades interactivament.
  - 2c) El sistema recorre totes les dades i les converteix en un format adequat.

## 6. Cas d'ús: Exportació dades preprocessades

- **Nom:** Exportació dades preprocessades
- **Actor:** Usuari
- **Precondició:** S'ha d'haver aplicat el preprocés de dades en el conjunt de dades importades al sistema.
- **Comportament:**
  - 1) L'usuari informa al sistema que vol exportar les dades preprocessades.
  - 2) El sistema demana un directori on exportar-les.
  - 3) El sistema demana un nom per al fitxer a exportar.
  - 4) El sistema converteix les dades preprocessades al format definit per al fitxer de dades preprocessades d'entrada.
  - 5) El sistema introdueix aquestes dades en el format adient en un fitxer amb el nom especificat
  - 6) El sistema exporta el fitxer al directori sol·licitat.
  - 7) Fi cas d'ús.
- **Error possible i cursos alternatius:**
  - 2a) El sistema detecta que el directori no existeix o no té permisos d'escriptura
  - 2b) Es torna a repetir el pas 2 del cas d'ús.
  - 3a) El sistema detecta que al directori on es vol exportar el fitxer, existeix un altre fitxer amb el mateix nom
  - 3b) Es torna a repetir el pas 3 del cas d'ús

## 7. Cas d'ús: Importació dades preprocessades

- **Nom:** Importació dades preprocessades
- **Actor:** Usuari
- **Precondició:** S'han d'haver exportat prèviament un conjunt de dades amb el format especificat.
- **Comportament:**
  - 1) El sistema pregunta a l'usuari en quin tipus de format vol proporcionar les dades.
  - 2) L'usuari selecciona el format "Fitxer dades preprocessades".
  - 3) El sistema demana el fitxer de dades preprocessades.
  - 4) L'usuari puja el fitxer de dades preprocessades.
  - 5) El sistema comprova que el format del fitxer sigui correcte.
  - 6) Fi del cas d'ús.
- **Errors possibles i cursos alternatius:**
  - 5a) El sistema retorna una excepció indicant "El fitxer adjuntat no té un format correcte."
  - 5b) Es torna a repetir des del pas 3 del cas d'ús.

## 8. Cas d'ús: Generar regles d'associació

- **Nom:** Generar regles
- **Actor:** Usuari
- **Precondició:** Les dades per generar les regles han d'estar preprocessades.
- **Comportament:**
  - 1) L'usuari informa al sistema que vol generar les regles.
  - 2) El sistema demana els paràmetres necessaris per la generació de regles: un conjunt de registres, suport i confiança mínims i el nombre màxim de regles que vols generar.
  - 3) L'usuari introdueix els paràmetres
  - 4) El sistema aplica l'algorisme per generar les regles d'associació.
  - 5) El sistema comprova que ha generat correctament les regles d'associació.
  - 6) El sistema retorna les regles generades en un fitxer.
  - 7) Fi cas d'ús.
- **Errors possibles i cursos alternatius:**
  - 2a) El sistema retorna una excepció si el suport mínim no està entre 0 i 1 "El suport mínim no és correcte".

2b) El sistema retorna una excepció si la confiança mínima no està entre 0 i 1 “La confiança mínima no és correcte”.

2c) El sistema retorna una excepció si el nombre màxim de regles és negatiu “El nombre de regles no és correcte”

## 9. Cas d'ús: Consulta de regles d'associació

- **Nom:** Consulta de regles d'associació

- **Actor:** Usuari

- **Comportament:**

- 1) L'usuari informa al sistema que vol consultar les regles d'associació generades per l'algorisme.

- 2) El sistema demana les característiques o els atributs que haurien de complir les regles.

- 3) L'usuari indica al sistema els atributs que l'interessen per la regla d'associació.

- 4) El sistema retorna la regla o les regles que compleixen els atributs indicats per l'usuari..

- 5) Fi cas d'ús.

- **Errors possibles i cursos alternatius:**

- 3a) L'usuari indica un atribut que no existeix en el sistema.

- 3b) El sistema retorna una excepció “No existeix algú dels atributs indicats”.

- 4a) El sistema retorna una excepció “No existeix cap regla d'associació que compleixi els atributs indicades”.

## 10. Cas d'ús: Eliminació de regles d'associació (opcional)

- **Nom:** Eliminació de regles d'associació

- **Actor:** Usuari

- **Precondició:** Hi hagin regles d'associació

- **Comportament:**

- 1) L'usuari informa al sistema que vol eliminar una regla d'associació.

- 2) El sistema elimina la regla seleccionada.

- 3) Fi cas d'ús.

- **Errors possibles i cursos alternatius:**

- 2a) El sistema retorna una excepció “No hi ha regles d'associació”.

## 11. Cas d'ús: Modificació de regles d'associació (opcional)

- **Nom:** Modificació de regles d'associació
- **Actor:** Usuari
- **Precondició:** Hi hagin regles d'associació
- **Comportament:**
  - 1) L'usuari informa al sistema que vol modificar una regla d'associació.
  - 2) El sistema proporciona un espai per canviar la regla.
  - 3) L'usuari modifica la regla en qüestió.
  - 4) El sistema comprova que el format de la regla sigui correcte.
  - 5) El sistema guarda el canvi.
  - 6) Fi cas d'ús.
- **Errors possibles i cursos alternatius:**
  - 2a) El sistema retorna una excepció "No hi ha regles d'associació".
  - 5a) El sistema retorna una excepció "La regla d'associació no està ben formulada".

## 12. Cas d'ús: Exportació de regles d'associació

- **Nom:** Exportació de regles d'associació
- **Actor:** Usuari
- **Precondició:** Les regles d'associació a exportar han estat generades per l'algorisme prèviament.
- **Comportament:**
  - 1) L'usuari informa al sistema que vol exportar unes regles d'associació un cop aplicat l'algorisme i modificades si calia/volia.
  - 2) El sistema demana un directori on exportar-les.
  - 3) El sistema demana un nom per al fitxer a exportar.
  - 4) El sistema converteix les regles d'associació al format definit per al fitxer de regles d'associació d'entrada.
  - 5) El sistema introdueix aquestes regles d'associació en el format adient en un fitxer amb el nom especificat
  - 6) El sistema inclou el format de preprocés aplicat a les dades dins del fitxer a exportar
  - 7) El sistema exporta el fitxer al directori sol·licitat.
  - 8) Fi cas d'ús.
- **Errors possibles i cursos alternatius:**
  - 2a) El sistema detecta que el directori no existeix o no té permisos d'escriptura
  - 2b) Es torna a repetir el pas 2 del cas d'ús.

- 3a) El sistema detecta que al directori on es vol exportar el fitxer, existeix un altre fitxer amb el mateix nom
- 3b) Es torna a repetir el pas 3 del cas d'ús

### 13. Cas d'ús: Importació de regles d'associació

- **Nom:** Importació de regles d'associació
- **Actor:** Usuari
- **Precondició:** Les regles d'associació a importar han d'haver estat exportats previament del sistema.
- **Comportament:**
  - 1) L'usuari informa al sistema que importar unes regles d'associació amb el seu procés.
  - 2) El sistema demana el fitxer de dades.
  - 3) L'usuari puja el fitxer en el sistema.
  - 4) El sistema importa el fitxer de regles d'associació.
  - 5) El sistema comprova que s'han importat correctament les regles d'associació.
  - 6) Fi cas d'ús.
- **Errors possibles i cursos alternatius:**
  - 5a) El sistema retorna una excepció "No s'han pogut importar les regles d'associació".
  - 5b) Es torna a repetir des del pas 2 del cas d'ús.

### 14. Cas d'ús: Validació de regles d'associació

- **Nom:** Validació de regles d'associació
- **Actor:** Usuari
- **Precondició:** S'han d'importar les regles d'associació i una entrada de dades preprocessades o dades no preprocessades amb les dades de preprocés corresponents.
- **Comportament:**
  - 1) L'usuari informa al sistema que vol validar les regles d'associació.
  - 2) El sistema comprova que el format d'entrada de dades correspon amb les regles d'associació.
  - 3) El sistema demana l'usuari el mínim suport i confiança.
  - 4) L'usuari entra al sistema el mínim suport i confiança.
  - 5) El sistema comprova que les regles d'associació es compleixen amb les noves dades.
  - 6) El sistema comprova que el suport i confiança obtinguts siguin més alts o iguals al mínim suport i confiança que ens ha donat l'usuari anteriorment.

7) El sistema retorna l'identificador, el suport i confiança de les regles que es compleixen.

8) Fi cas d'ús.

• **Errors possibles i cursos alternatius:**

4a) El sistema retorna una excepció si el suport mínim no està entre 0 i 1 "El suport mínim no és correcte".

4b) El sistema retorna una excepció si la confiança mínima no està entre 0 i 1 "La confiança mínima no és correcte".

## 15. Cas d'ús: Conversió de regles d'associació (*opcional*)

• **Nom:** Conversió de regles d'associació (*opcional*)

• **Actor:** Usuari

• **Precondició:** El sistema ha de tenir unes regles d'associació ja processades.

• **Comportament:**

1) L'usuari indica al sistema que vol convertir les regles d'associació.

2) El sistema dona un llistat d'opcions per convertir les regles:

a) PDF

b) Excel

c) ...

3) L'usuari escull una de les opcions.

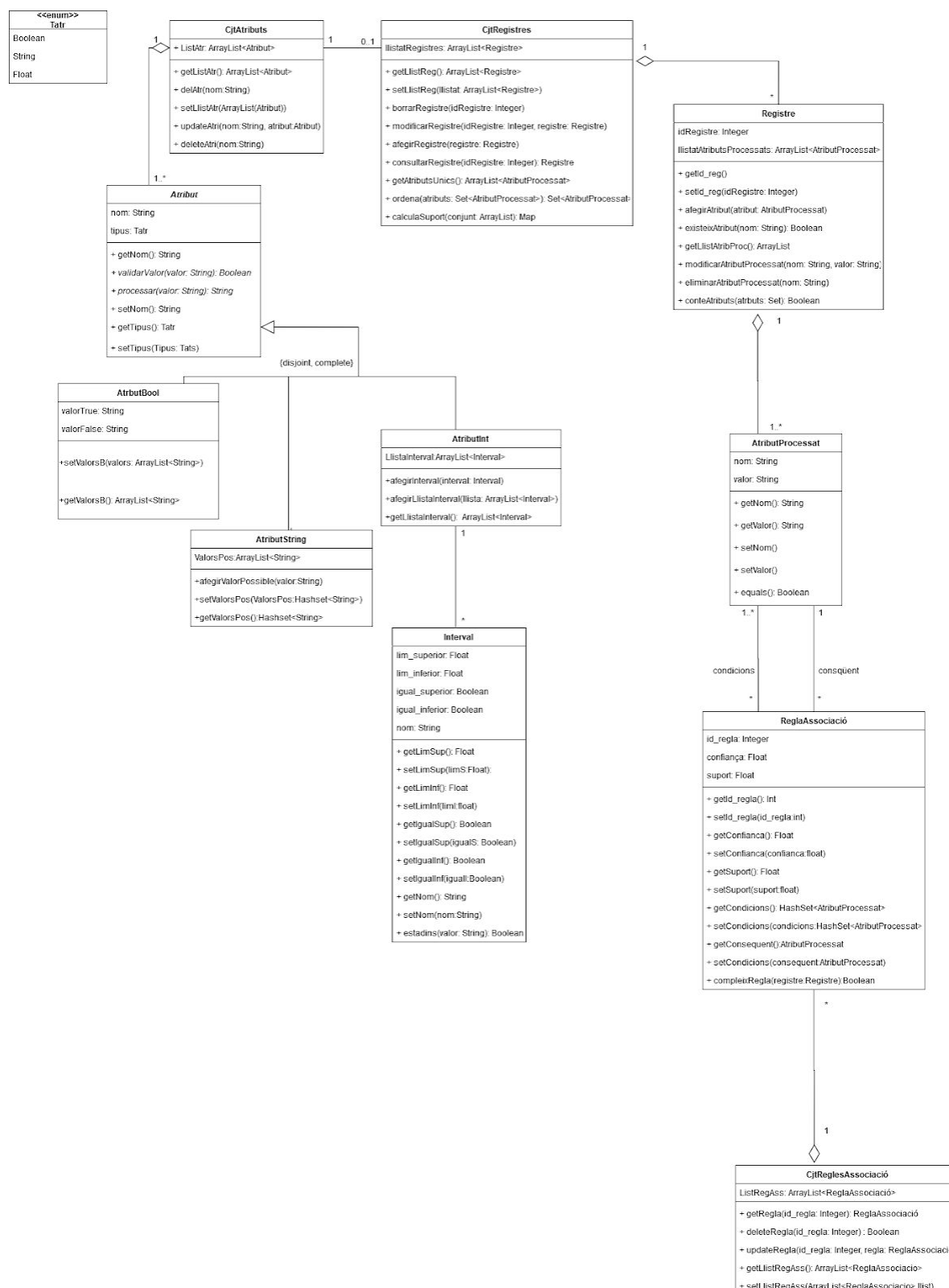
4) El sistema proporciona a l'usuari un arxiu amb les dades convertides.

5) Fi cas d'ús.

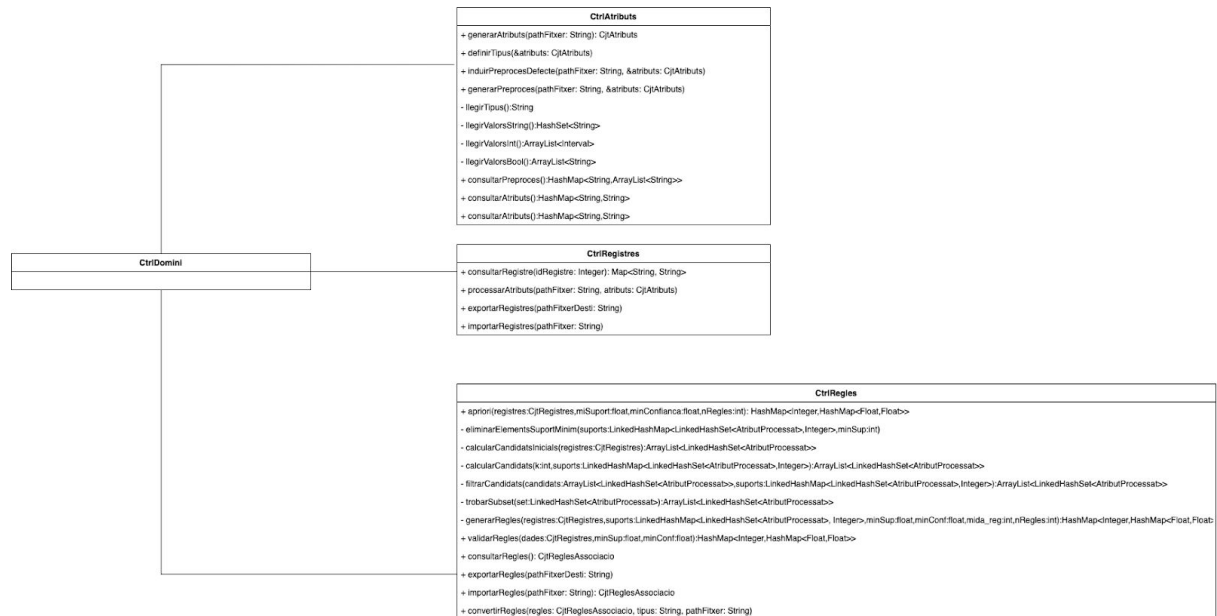
• **Errors possibles i cursos alternatius:**



# Diagrama de classes de l'especificació del sistema



(Es pot trobar aquesta imatge al directori **DOCS** de l'entrega).



(Es pot trobar aquesta imatge al directori **DOCS** de l'entrega).