

Projecte final de carrera

**Sistema intel·ligent d'ajuda a la decisió pel control
remot de tecnologia de membranes per la
depuració d'aigües.**

Afra Sabrià i Villodres

2012

Índex

1. Introducció, motivacions, propòsit i objectius del projecte.....	7
1.1. Grup de recerca LEQUIA	7
1.2. El projecte COLMATAR+	9
1.3. Objectius del Projecte Final de Carrera.....	11
1.4. La planta pilot.....	12
1.4.1. Descripció de la planta pilot.....	12
1.4.2. Descripció de les diferents unitats	15
2. Estudi de viabilitat.....	21
3. Metodologia	23
4. Planificació	25
5. Marc de treball i conceptes previs	27
5.1. Marc de treball.....	27
5.2. El sistema informàtic existent	29
5.2.1. Aplicació Java <i>colmatar</i>	30
5.2.2. Aplicació web	38
5.2.3. Aplicació pel disseny de les regles de control	44
5.2.4. Base de dades.....	45
5.3. Grup MBR del LEQUIA	47
5.4. Què es vol fer?	48
6. Requisits del sistema.....	51

7. Estudis i decisions.....	53
8. Anàlisi i disseny del sistema	55
8.1. Aplicació Colmatar	55
8.1.1. Anàlisi de com està estructurada	55
8.1.2. Disseny.....	59
8.1.3. Diagrama de dependències.....	59
8.1.4. Diagrames de classes.....	61
8.2. Aplicació web	68
8.2.1. Anàlisi	68
8.2.2. Disseny	69
8.3. Base de dades.....	73
8.4. Disseny de les regles de control.....	73
8.4.1. MBR Control Aire.....	74
8.4.2. StartUp Kubota Module	76
9. Implementació i proves.....	79
9.1. MBR Control Aire.....	79
9.1.1. Disseny de la regla de control	80
9.1.2. Proves d'execució.....	88
9.2. StartUp Kubota Module	89
9.2.1. Disseny de la regla de control	89
9.2.2. Proves d'execució.....	98
10. Implantació i resultats.....	99

10.1. MBR Control Aire Membranes	99
10.2. StartUp Kubota Module	101
11. Conclusions.....	103
12. Treball futur.....	105
13. Bibliografia	107
14. Annexos.....	109
15. Manual d'usuari i/o instal·lació.....	111

1. Introducció, motivacions, propòsit i objectius del projecte

Fa tres anys que formo part del grup de recerca LEQUIA (Laboratori d'Enginyeria Química i Ambiental, <http://lequia.udg.cat>). Vaig tenir aquesta oportunitat a través d'una oferta a la borsa de treball de la Universitat de Girona.

Des de la meua incorporació formo part del grup encarregat de la recerca mitjançant bioreactors de membrana (MBR) per al tractament d'aigües i la seva reutilització, anomenat grup MBR o de membranes (www.colmatar.es).

Primerament vaig estar un any com a becària i des de llavors sóc tècnica de suport contractada al projecte COLMATAR+ (CTM2009-14742-C02-01).

1.1. Grup de recerca LEQUIA

LEQUIA és un grup de recerca adscrit a l'Institut de Medi Ambient de la Universitat de Girona integrat en la Xarxa d'Innovació Tecnològica del CIDEM (actualment ACC1Ó) des de l'any 2001. Actualment forma part de la nova marca d'agents de transferència tecnològica TECNIO. També, recentment ha obtingut la renovació com a Grup de Recerca Consolidat de la Generalitat de Catalunya que gestionen l'AGAUR i Talència.

L'equip humà està format per un equip estable de set professors titulars d'enginyeria química que coordinen més de trenta col·laboradors, principalment, estudiants de doctorat i tècnics de suport especialitzats. Una de les peculiaritats del grup és que treballa de manera multi disciplinar, ja que en els projectes portats a terme intervenen ambientòlegs, químics, biòlegs, enginyers i informàtics. Aquesta recerca té lloc a la Facultat de Ciències (Campus Montilivi) i al Parc Científic i Tecnològic de la Universitat de Girona (Edifici Jaume Casademont), on el grup té distribuïts els laboratoris de recerca i docència i els despatxos dels investigadors, estudiants de doctorat i tècnics de suport (330 m²).

En la formació de postgrau, el grup participa en el Màster oficial de Ciència i Tecnologia de l'Aigua de la Universitat de Girona i els estudiants de doctorat estan adscrits al programa de Doctorat de Ciències Experimentals i Sostenibilitat amb reconeixement de menció de qualitat.

L'activitat investigadora del grup s'estructura en les següents línies d'expertesa:

- Disseny, operació i control de processos avançats pel tractament biològic d'aigües residuals urbanes i industrials.

- Desenvolupament i aplicació de sistemes d'ajuda a la decisió en dominis mediambientals.
- Estudi de processos d'adsorció/oxidació per al tractament d'efluents líquids i gasosos.

Aquesta activitat es porta a terme a través de:

- Projectes de recerca, desenvolupament i innovació tecnològica
- Contractes i/o convenis per la transferència de resultats d'investigació a les empreses i administracions públiques
- Formació d'investigadors i professionals del sector de tractament d'aigües residuals.

Per respondre adequadament a les necessitats dels clients, el grup LEQUIA compta amb un equip humà multi disciplinar altament qualificat i amb els mitjans tècnics necessaris per la realització d'activitats d'R+D+i. El grup disposa de diverses instal·lacions amb plantes pilots i equipament analític i de control necessari per dur a terme els diferents estudis i les activitats de recerca bàsica i aplicada que es duen a terme a partir de projectes competitius amb o sense participació empresarial i de contractes/convenis R+D+i amb empreses privades i administracions públiques. D'altra banda, el grup disposa de tots els recursos computacionals (hardware i software) necessaris per les tasques de modelat matemàtic avançat, simulació i control de processos.

El grup de recerca LEQUIA desenvolupa serveis d'R+D+i en resposta a les demandes d'un gran nombre d'empreses i d'entitats responsables de la gestió de l'aigua i els residus. Aquests serveis s'han realitzat amb PIMEs (Petites i Mitjanes Empreses) i grans empreses dels sectors agroalimentari, quimicofarmacèutic, empreses de serveis del sector del tractament d'aigües residuals, i amb administracions locals i nacionals.

La oferta científica i tecnològica d'aquests serveis es basa en el desenvolupament i aplicació d'avançades metodologies d'anàlisi experimental i modelat matemàtic per optimitzar el disseny, operació i control de les tecnologies de tractament d'aigües, fangs i residus sòlids així com en aplicacions telemàtiques en enginyeria ambiental i la gestió integrada de l'aigua.

Cal destacar la recent concessió de la patent nacional "*Procedimiento automatizado de control en tiempo real de un biorreactor de membranas (MBR) y sistema de control correspondiente*", la titularitat de la qual està compartida al 50% amb l'empresa OHL Medio Ambiente INIMA.

Dels serveis a l'empresa que ofereix el grup LEQUIA, els més rellevants són:

- Desenvolupament de nous processos de tractament d'efluents (nitritació parcial, anammox, membranes, microbial fuel cells, adsorció/oxidació).
- Millora de la gestió dels processos de tractament en funcionament.
- Modelització i simulació de processos (SAD - Sistema d'Ajuda a les Decisions).
- Avaluació de la biodegradabilitat d'efluents. Assaigs en planta pilot.
- Aplicació d'eines de gestió ambiental.
- Estudis de processos per l'aplicació de tecnologies més netes.

Actualment dono suport al projecte COLMATAR+ (CTM2009-14742-C02-01, extensió del projecte COLMATAR amb referència DPI2006 de 15707 C02) on es van definir les bases d'un sistema de suport a la decisió pel control de tecnologia de membranes.

Dins aquest projecte és on s'ha realitzat aquest Projecte de Final de Carrera.

1.2. El projecte COLMATAR+

Una de les principals línees d'investigació del LEQUIA dels darrers anys és la millora del tractament d'aigües mitjançant tecnologia de membranes dins del marc dels projectes COLMATAR i COLMATAR+ finançats pel Ministerio de Ciencia e Innovación. L'objectiu principal d'aquests projectes és el disseny d'un sistema d'ajuda a la decisió (SAD) basat en el coneixement pel control i la supervisió integrada de bioreactors de membrana (BRM) per la depuració d'aigües residuals amb exigències de reutilització.

La Directiva Marc de l'Aigua potencia la gestió integrada de l'aigua, la seva reutilització i el desenvolupament de noves metodologies i eines d'ajuda a la presa de decisions. A la vegada, amb el tractament descentralitzat es preveu un augment progressiu del número de petites Estacions Depuradores d'Aigües Residuals (EDARs), amb les particularitats inherents d'exigència de fiabilitat i automatisme en contrapartida a un baix nivell de mà d'obra.

Els bioreactors de membranes (BRM) estan guanyant terreny en el tractament d'aigües residuals, principalment en estacions petites amb elevades exigències de qualitat de l'aigua tractada per la seva reutilització. No obstant, no existeixen estudis que relacionin el tipus i freqüència del rentat de les membranes amb els principals paràmetres d'operació del procés biològic i especialment amb els desequilibris de tipus microbiològic.

De forma genèrica, un bioreactor de membranes es pot definir com un reactor de fangs actius en el qual se li ha reemplaçat l'etapa de sedimentació secundària per una etapa de filtració

mitjançant membranes de micro o ultrafiltració. Les membranes, que estan formades per un material porós amb diàmetres de porus petit (entre 0.4 i 0.01 μm), no tan sols substitueixen el decantadors secundaris (els microorganismes ja no han de decantar) sinó que la qualitat de la sortida que proporcionen és equiparable a la dels tractaments terciaris convencionals, disminuint alhora considerablement la mida dels bioreactors. Per poder seguir l'evolució del procés de filtració és necessari controlar una sèrie de paràmetres clau descrits a continuació:

- **Flux (J):** Quantitat de material que passa a través d'una unitat de superfície de membrana per unitat de temps. Unitats (SI): $\text{m}^3 \cdot \text{m}^{-2} \cdot \text{s}^{-1}$ o bé, $\text{m} \cdot \text{s}^{-1}$ o bé, LMH.
- **Pressió transmembrana (TMP, abreviatura de l'anglès Transmembrane Pressure) :** paràmetre que s'utilitza per expressar l'energia necessària per a la filtració, també anomenada caiguda de pressió. Aquest paràmetre es calcula mitjançant la diferència entre la pressió estàtica sense filtració i la pressió dinàmica durant la filtració (descomptant-hi les pèrdues degudes a les canonades).

Els Sistemes d'Ajuda a la Decisió (SAD) s'han mostrat molt vàlids i adequats per la seva aplicació a diferents camps mediambientals, entre ells el procés de tractament biològic d'aigües residuals. Existeixen metodologies validades pel desenvolupament de SAD, que inclouen l'ús de tècniques avançades d'intel·ligència artificial i mineria de dades per l'adquisició i estructuració del coneixement general i específic de processos ambientals.

El LEQUIA i el Centro de Estudios e Investigaciones Técnicas de Guipúzcoa (CEIT, www.ceit.es) tenen una ampla i reconeguda experiència en el camp del tractament biològic d'aigües residuals (tant en planta pilot com en instal·lacions reals), centrant-se més en l'ús de tècniques d'intel·ligència artificial i el desenvolupament de SAD els investigadors de LEQUIA-UdG, i més en aspectes de control robust, modelització i tecnologia de membranes els investigadors de CEIT.

Aquesta experiència està avalada pels seus múltiples projectes i publicacions internacionals en aquest camp, en els que precisament s'hi ha constatat la seva validesa en diferents hipòtesis de partida del projecte.

Per aquest motiu, s'ha decidit de desenvolupar l'anàlisi, disseny, implementació i validació d'un SAD intel·ligent pel control remot i la supervisió de bioreactors de membrana.

Dins el projecte COLMATAR+ hi havia la necessitat de disposar d'un aplicatiu informàtic que garanteixi l'adquisició i monitorització, el control robust i autònom de certes variables, així

com la detecció d'errors, alarmes i fallades i que alhora permeti la seva supervisió experta i remota dels bioreactors de membrana. Així doncs, calia dotar al sistema d'un registre d'actuacions, per posteriorment analitzar-lo i permetre la reutilització en el futur. Del coneixement adquirit per poder afrontar problemes similars i validar l'aplicatiu en planta pilot a escala industrial.

1.3. Objectius del Projecte Final de Carrera

El principal objectiu del projecte consisteix en desenvolupar **l'anàlisi, disseny, desenvolupament i implementació d'un sistema d'ajuda a la decisió (SAD), basat en el coneixement, pel control remot i la supervisió de la operació integrada d'estacions depuradores BRM (bioreactor de membranes)** per la depuració d'aigües residuals amb exigències de qualitat de reutilització de l'aigua tractada.

La idea és obtenir un prototip informàtic que garanteixi la monitorització i el control robust i autònom, així com la detecció d'errors, i que a la vegada permeti la supervisió remota de l'operació integrada del procés biològic i el tractament fisicoquímic de les membranes mitjançant un conjunt de mòduls implementats com a sistemes basats en regles.

A més de l'adquisició de coneixement general del procés, pel desenvolupament del sistema d'ajuda a la decisió es plantegen una sèrie d'experiments en una planta pilot a escala industrial que ens permetrà determinar les condicions ideals i extreure'n coneixement empíric i específic que ens permetrà relacionar paràmetres de control biològic amb la operació de la unitat de membranes.

Es tracta d'un enfocament multi disciplinar (procés biològic, modelització, automatització, control, intel·ligència artificial, mineria de dades, telegestió, etc.) i multi objectiu (minimització de costos, fiabilitat del procés, qualitat de l'aigua, manteniment de membranes, etc.).

La principal finalitat del projecte és el desenvolupament, implementació i validació d'un Sistema d'Ajuda a la Decisió, que integri diferents eines complementàries, per la gestió integrada de sistemes de sanejament basats en BRM, una combinació de tractaments biològics i fisicoquímics, que tindrà com a principals metes:

- Augmentar la fiabilitat operacional,
- Estalviar costos operacionals i energètics,
- Mantenir o millorar la qualitat del efluent per reutilització (major eficiència), i,

- Aprendre automàticament i reutilitzar el coneixement adquirit.

1.4. La planta pilot

En aquest apartat es descriuen les característiques principals de la planta pilot de depuració d'aigües residuals mitjançant tecnologia BRM en la que s'ha desenvolupat l'aplicació del meu projecte.

1.4.1. Descripció de la planta pilot

El grup disposa de varies plantes pilot en les que es realitzen les proves pertinents. Bàsicament totes disposen del mateix sistema informàtic, però cadascuna amb les seves particularitats ja que totes són diferents, tenen diferents esquemes de la planta i per tant diferents sensors.

La planta pilot utilitzada per realitzar aquest projecte es troba situada a l'Estació Depuradora d'aigües residuals de Castell Platja d'Aro, comarca del Baix Empordà. Tot i que el grup disposa d'altres plantes pilot, el projecte es centrarà bàsicament en aquesta ja que és on s'han realitzat més proves del mòdul automàtic desenvolupat, tot i que també s'ha fet alguna prova en una altra planta.

A continuació es pot veure la vista aèria de l'estació depuradora d'aigües residuals de Castell – Platja d'Aro, on hi ha situada la planta pilot.



Figura 1. Vista aèria de l'EDAR i situació de la planta pilot

A la imatge següent podem veure el contenidor on s'ubica la planta pilot.



Figura 2. Contenidor de la planta pilot

En aquesta Estació Depuradora es tracten les aigües residuals de tres poblacions: Sant Feliu de Guíxols, Santa Cristina d'Aro i Castell-Platja d'Aro. Els diferents destins de l'aigua que surt d'aquesta depuradora són el Ridaura i el mar posteriorment. Part d'aquesta aigua tractada es reutilitzada en regs agrícoles i també en el Golf d'Aro i el Golf Costa Brava.

El tractament que segueix aquesta EDAR convencional és a través de fangs activats. Aquests fangs que es generen durant el procés de tractament de les aigües és reutilitzen amb fins agrícoles. Aquesta EDAR ocupa un espai de 3,5 Ha i està dissenyada per tractar un cabal de 35.000 m³/dia. L'equivalent en habitants del que es tracta en aquesta planta és 175.000 habitants equivalents.

La planta pilot BRM, però, només tracta l'aigua provinent de Santa Cristina d'Aro, un 95% de l'aigua residual d'aquesta població és d'origen urbà domèstic i la resta és industrial. El cabal que es tracta és de 168L/h.

A la imatge següent veiem l'interior de la planta pilot.



Figura 3. Imatge de l'interior de la planta pilot

En el següent esquema es pot veure les diferents parts que formen la planta pilot de Castell-Platja d'Aro.

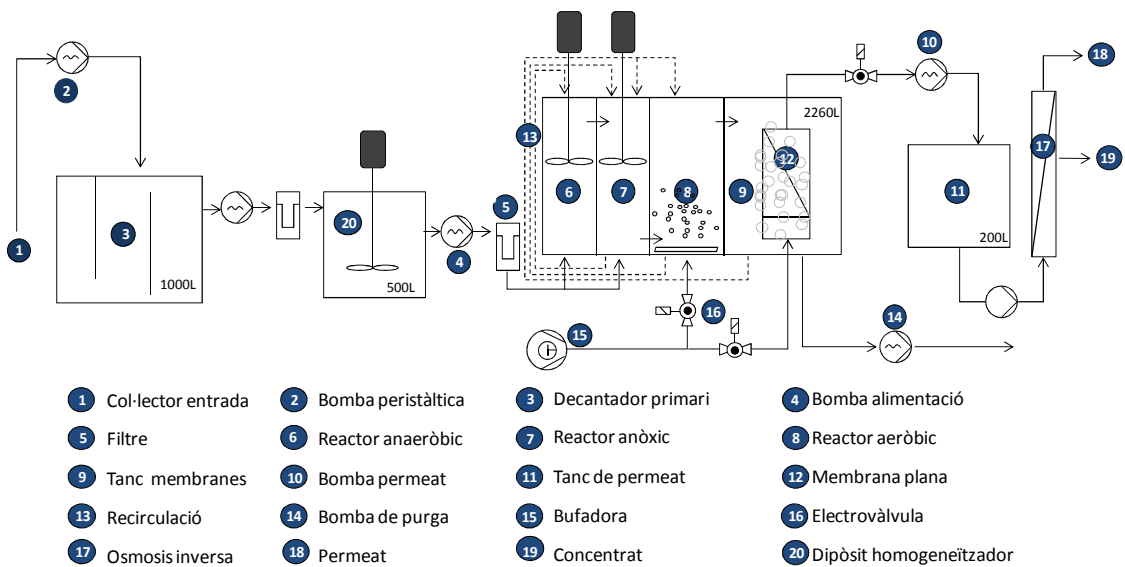


Figura 4. Esquema de la planta pilot

La bomba peristàltica (2) es troba situada a l'arqueta on arriben les aigües residuals del col·lector del municipi de Santa Cristina d'Aro (1) i aquesta impulsa cap a la planta un cabal d'aigua que està condicionat pel nivell en què es trobi el tanc homogeneïtzador (20). Abans d'entrar al tanc homogeneïtzador, passa per un primer filtre o tamís, d'una mida de porus d'1mm. Un cop al tanc d'homogeneïtzador, una segona bomba (4) impulsa un cabal d'aigües residuals homogeneïtzades cap al bioreactor, fent-lo passar abans per un segon filtre (5). El bioreactor té una configuració UCT (*University of Cape Town*) i és on tenen lloc els processos biològics i on es troben les membranes (9 i 12). L'aigua que s'obté de la microfiltració amb membranes s'impulsa cap al tanc de permeat (11) i aquest alimenta el mòdul d'osmosi inversa (17). Hi ha diferents recirculacions necessàries perquè es duguin a terme els diferents processos biològics i una purga de fangs per a intentar mantenir constant l'edat cel·lular dels fangs.

1.4.2. Descripció de les diferents unitats

En aquest apartat es descriuen les diferents unitats que componen la planta pilot.

1.4.2.1. Dipòsit homogeneïtzació

El dipòsit d'homogeneïtzació es troba situat a la part esquerra de la planta pilot. El volum del dipòsit és de 500L dins del qual es troba un agitador per mantenir en constant homogeneïtzació l'aigua del seu interior. El tanc està destinat a oferir per un costat un volum mínim necessari perquè el bioreactor pugui treballar a un cabal constant i l'altra funció del dipòsit és amortir els pics punta de concentració de nutrients.

En el dipòsit trobem també un transmissor de nivell per saber el nivell del dipòsit en qualsevol moment. El dipòsit disposa d'un sobreeixidor per si el nivell d'aigua de l'interior supera el valor màxim de capacitat.

Previ al dipòsit d'homogeneïtzació es troba un filtre de bossa amb porus de mida de 1mm, la funció del qual es retenir aquelles partícules de mida superior a les de la mida del porus. I posterior al dipòsit es troba un filtre tamís amb porus de mida de 0.5 mm. A la imatge es poden veure representats els dos tipus de filtres.



Figura 5. Filtres

Tots dos filtres estan controlats per un transmissor de pressió per indicar la saturació d'ambdós filtres.

1.4.2.2. Bioreactor

Posterior al dipòsit d'homogeneització es troba el bioreactor d'una capacitat màxima de 2260L, el qual està dividit en 4 tancs independents: anaeròbic, anòxic, aeròbic i membranes, connectat entre ells per mitjà de recirculacions o per mitjà de passos entre reactors contigus.

El bioreactor disposa d'un sobreexidor (situat al tanc de membranes) per extreure l'aigua que pugui superar el valor de capacitat màxima. Mitjançant l'esquema es pot observar les diferents parts del bioreactor.

A 320 L	B 320 L	C 520L	D 1100 L
-------------------	-------------------	------------------	--------------------

Figura 6. Esquema bioreactor amb configuració UCT

Esquema del bioreactor. (A) reactor anaeròbic. (B) reactor anòxic. (C) reactor aeròbic. (D) tanc de membranes.

Les principals característiques físiques i instrumentals de cada reactor són les següents:

- Reactor anaeròbic (A): Està situat a la part esquerra del bioreactor . Aquest reactor té 320L de capacitat màxima (un 14% del volum total) i està equipat amb un agitador per mantenir en suspensió la biomassa present en el seu interior. Aquest reactor a més, està equipat amb una sonda de sòlids en suspensió i una sonda d'amoni.
- Reactor anòxic (B): Aquest reactor té 320L de capacitat màxima (un 14% del volum total) i està equipat amb un agitador per mantenir en suspensió la biomassa present en el seu interior. A més també disposa d'una sonda RedOx per saber un qualsevol moment el valor del potencial oxidació-reducció.
- Reactor aeròbic (C): Aquest reactor té una capacitat màxima de 520L (un 23% del volum total). Està equipat amb uns difusors d'aire, els quals oferiran un cabal d'aire per mantenir una concentració d'oxigen estipulada prèviament. Dispos a més, d'una sonda de pH, de temperatura i d'oxigen dissolt.
- Reactor de membranes (D) : Posterior al reactor aeròbic trobem el tanc de membranes que té un volum de 1100L (49% del bioreactor) destinat al procés d'ultrafiltració. En aquest tanc es troba el cartutx de membranes. Hi ha diferents tipus de membranes, tot i que les més utilitzades actualment són les dues següents: de fibra buida i de làmina plana. Les utilitzades en la planta COLMATAR són del tipus de fibra buida. Són de la casa comercial KUBOTA i es caracteritzen per tenir uns porus mida 0.4 µm i una àrea total de membranes de 8 m².
Aquest tanc està equipat amb un sonda d'oxigen dissolt, una sonda d'amoni i una sonda de sòlids en suspensió. A més també es troba un transmissor de nivell per saber el nivell del bioreactor en qualsevol moment.

1.4.2.3. Mòdul d'osmosi inversa

El mòdul d'OI (Osmosi inversa) està alimentat per l'aigua provinent del MBR, aquesta és impulsada per una bomba prèvia i després per una bomba d'alta pressió cap a la membrana d'osmosi inversa, un cop ha passat la membrana el concentrat pot ser recirculat cap al dipòsit d'alimentació de l'OI o cap a desguàs. El permeat es dirigeix cap a un dipòsit de rentat i el sobreexigit cap a la sortida de planta pilot.

També posseeix un sistema automàtic de rentat amb aigua permeada d'OI, que es posa en funcionament cada vegada que el sistema s'atura per qualsevol esdeveniment. Equipat amb sondes de conductivitat, pH i temperatura. Així com amb cabalímetres d'alimentació i permeat. Per comptabilitzar la pèrdua de pressió hi ha transductors de pressió instal·lats a l'entrada i sortida de la membrana OI.

1.4.2.4. Panell de control PLC – SCADA

A partir del panell de control es poden modificar, observar i controlar tots els paràmetres, instruments i, alarmes que hi pugui haver o es puguin determinar en el procés amb tecnologia BRM del tractament d'aigües residuals.

A continuació podem veure el panell de control i l'SCADA (*Supervisory Control And Data Adquisition*) on es veu representat el mòdul d'osmosi inversa.



Figura 7. Panell de Control

A la següent imatge es pot veure la pantalla del SCADA de la planta pilot.

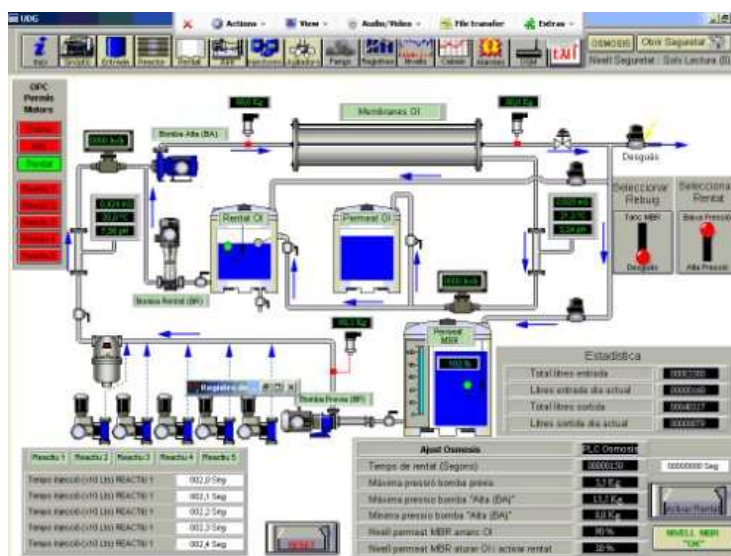


Figura 8. SCADA

El mòdul de control automàtic a desenvolupar en aquest PFC basat en regles de control permetrà realitzar, segons unes condicions determinades, canvis en els valors de les variables d'actuació que hi ha disponibles a la planta. Això ens permet realitzar canvis a la planta que normalment faria l'usuari encarregat de l'estudi, i que al tenir el mòdul de control automàtic els canvis es realitzen automàticament sense que l'usuari hi hagi d'intervenir.

Per exemple una de les regles de control que es desenvoluparà en aquest projecte ha de fer varies comprovacions. Una de les moltes comprovacions que farà serà llegir el valor de TMP (pressió transmembrana) corresponent a l'aparell amb id 255, si el valor d'aquest és més petit o igual que -170 mbar ($TMP \leq -170$), llavors el control ha de demanar la desactivació del sistema de control automàtic de la regla corresponent.

2. Estudi de viabilitat

Els responsables del projecte van decidir dur a terme la part informàtica del projecte, ja que era una part molt important pels objectius finals dels projectes COLMATAR i COLMATAR+.

En aquest apartat no hi he intervingut personalment.

3. Metodologia

Aquest projecte s'ha desenvolupat a partir d'un altre projecte inicial (COLMATAR) on es va definir un pla de treball que es pot desglossar en un conjunt de tasques associades a cada un dels objectius concrets del projecte. Per cada objectiu s'han determinat diferents tasques, especificant les fites que es preveuen assolir i el grup i les persones responsables i involucrades per dur-les a terme.

La major part de les tasques són del tipus d'investigació i desenvolupament però també se n'inclouen algunes de divulgació i explotació. Totes elles per tal de dur a terme d'adquisició del coneixement adquirit i executar els desenvolupaments de tècniques relacionades amb la tecnologia MBR i la seva aplicació pràctica d'optimització i validació d'un sistema SAD pel control integrat.

La metodologia utilitzada per desenvolupar la part informàtica segueix els punts més importants de les metodologies tradicionals d'anàlisi i desenvolupament de software que són:

- Estudi de l'aplicació: estudiar de què disposem, quin o quins softwares tenim ja desenvolupats i ens poden ser útils a l'hora de desenvolupar aquest nou mòdul del projecte.
- Anàlisi dels requisits: analitzar si amb el que ja disposem podem desenvolupar el nou mòdul o bé hem de desenvolupar noves aplicacions per tal de poder-ho integrar.
- Model de dades: estudiar de quin model de dades disposem i què necessitem per al desenvolupament del nou mòdul.
- Implementació: un cop realitzats els estudis previs del que ja disposem i del que cal modificar o desenvolupar de nou, implementar el nou mòdul.
- Proves: un cop implementat el mòdul realitzar les proves que ens permetin comprovar que el mòdul que hem dissenyat actua segons el que s'espera.
- Explotació: si les proves de validació realitzades han sigut satisfactòries, es posa en funcionament el mòdul implementat quan el grup de MBR decideix realitzar aquell experiment.

5. Marc de treball i conceptes previs

Abans de començar a desenvolupar el projecte vaig fer un estudi del que ja hi havia fet. En aquest estudi es van analitzar els següents apartats.

5.1. Marc de treball

En el marc del projecte COLMATAR es va desenvolupar una aplicació en llenguatge de programació Java que s'encarregava de gestionar les dades dels PLC (*Programmable logic controller*) que rebia a través d'un servidor OPC (*OLE(Object Linking and Embedding) for Process Control*) i llavors poder guardar-les dins una base de dades, en aquest cas MySQL.

Més endavant es va implementar una aplicació web per tal de poder interactuar amb l'usuari, aquesta està desenvolupada amb JavaScript i AJAX. La decisió de desenvolupar cada aplicació en aquests llenguatges i la manera d'estructurar-ho va ser decisió del desenvolupador que hi havia abans al grup LEQUIA.

En el moment de la meua incorporació al LEQUIA i l'assignació del projecte, ja hi havia una primera idea de com es podria desenvolupar aquest nou mòdul. D'aquesta manera també s'utilitza una aplicació ja creada que permet transformar cada regla de control que es dissenya en un fitxer XML.

Per tant aprofitant aquesta aplicació, a partir d'aquí es decideix que el nou mòdul es desenvoluparà interpretant aquest fitxer XML que es crea.

A la figura 10 podem veure un esquema de la comunicació/relació entre les aplicacions existents. La part MBR seria la planta pilot i els seus sensors connectats al PLC i SCADA on s'encarreguen de rebre les dades corresponents. Aquestes dades es traspassen a través d'un servidor OPC que en l'esquema estaria representat pel SIGNAL PROCESSING i el CONTROL. Aquestes tasques les fa l'aplicació en Java *colmatar*. La part de SUPERVISION representaria l'aplicació web que és la que interactua l'usuari USER.

Tal i com es veu en l'esquema la part de SIGNAL PROCESSING, CONTROL i SUPERVISION estan integrades dins del Mini PC. I a part tindríem, l'SCADA, el PLC, la planta MBR i l'USER totes elles per separat però comunicant-se tal i com indica l'esquema.

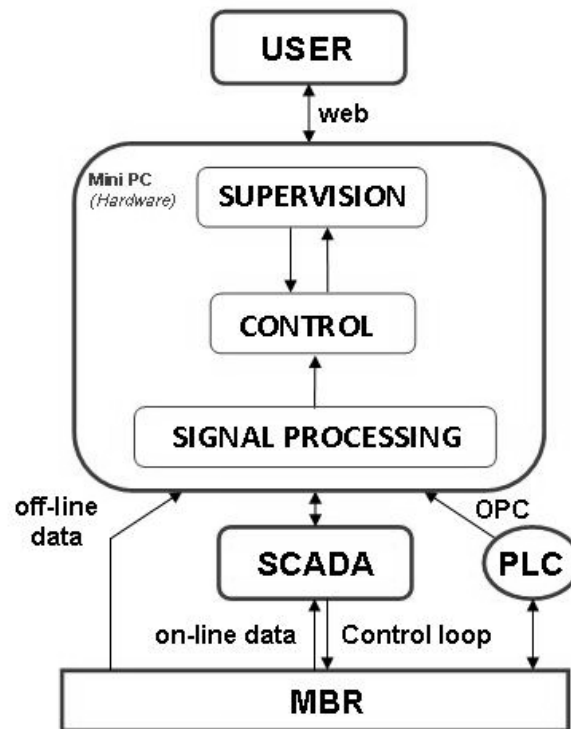


Figura 10. Esquema de comunicacions

El servidor OPC de la planta s'encarrega de tenir guardats els aparells mitjançant la seva adreça de memòria, que ens proporciona l'instal·lador del PLC.

Per tant disposem d'un fitxer amb extensió .cdm on hi ha descrits cada aparell del que rebem dades a través del PLC i que el servidor OPC llegeix i interpreta i s'encarrega de fer de pont amb la nostra aplicació *colmatar*.

Per cada planta pilot tindrem aquest fitxer .cdm , que veiem a continuació, d'acord amb els aparells de què disposi.

Nombre	Tipo	Dirección	Comentario
AutoGen_0_00	Punto	NuevoPLC1/0.0/1/B	Botonera
AutoGen_0_01	Punto	NuevoPLC1/0.1/1/B	Termic airejador entrada
AutoGen_0_02	Punto	NuevoPLC1/0.2/1/B	Botonera
AutoGen_0_03	Punto	NuevoPLC1/0.3/1/B	Termic airejador anaerobic
AutoGen_0_04	Punto	NuevoPLC1/0.4/1/B	Botonera
AutoGen_0_05	Punto	NuevoPLC1/0.5/1/B	Termic airejador desnitrificació
AutoGen_0_06	Punto	NuevoPLC1/0.6/1/B	Botonera Bomba Alimentació Planta
AutoGen_0_07	Punto	NuevoPLC1/0.7/1/B	Termic Bomba Alimentació planta
AutoGen_0_08	Punto	NuevoPLC1/0.8/1/B	Botonera Bomba
AutoGen_0_09	Punto	NuevoPLC1/0.9/1/B	Termic Bomba Alimentació reactor
AutoGen_0_10	Punto	NuevoPLC1/0.10/1/B	Botonera
AutoGen_0_11	Punto	NuevoPLC1/0.11/1/B	Termic recirculadora principal
AutoGen_0_12	Punto	NuevoPLC1/0.12/1/B	Botonera
AutoGen_0_13	Punto	NuevoPLC1/0.13/1/B	Termic recirculadora auxiliar 1
AutoGen_0_14	Punto	NuevoPLC1/0.14/1/B	Botonera
AutoGen_0_15	Punto	NuevoPLC1/0.15/1/B	Termic recirculadora auxiliar 2
AutoGen_1_11	Punto	NuevoPLC1/1.11/1/B	Botonera Sosa
AutoGen_1_12	Punto	NuevoPLC1/1.12/1/B	Termic Acid/Sosa
AutoGen_1_13	Punto	NuevoPLC1/1.13/1/B	Botonera Antiescumant
AutoGen_1_14	Punto	NuevoPLC1/1.14/1/B	Termic Sosa
AutoGen_1_15	Punto	NuevoPLC1/1.15/1/B	Fallo Electric
AutoGen_2_00	Punto	NuevoPLC1/2.0/1/B	Botonera Sosa/acid
AutoGen_2_01	Punto	NuevoPLC1/2.1/1/B	Termic Antiescumant
AutoGen_2_02	Punto	NuevoPLC1/2.2/1/B	Nivell Sosa
AutoGen_2_03	Punto	NuevoPLC1/2.3/1/B	Nivell Antiescumant
AutoGen_2_04	Punto	NuevoPLC1/2.4/1/B	Nivell Sosa/acid
AutoGen_2_06	Punto	NuevoPLC1/2.6/1/B	ON=Valvula Tancada
AutoGen_2_07	Punto	NuevoPLC1/2.7/1/B	ON=EV permeat_tanc
AutoGen_2_08	Punto	NuevoPLC1/2.8/1/B	ON=Valvula Tancada
AutoGen_2_09	Punto	NuevoPLC1/2.9/1/B	ON=EV Tanc Permeat Obert
AutoGen_2_10	Punto	NuevoPLC1/2.10/1/B	ON=EV Entrada Permeat Tancada
AutoGen_2_11	Punto	NuevoPLC1/2.11/1/B	ON=Valvula Rentat Oberta
AutoGen_2_12	Punto	NuevoPLC1/2.12/1/B	ON=Valvula Rentat Tancada
AutoGen_2_13	Punto	NuevoPLC1/2.13/1/B	ON=EV recircula_permeat
AutoGen_2_14	Punto	NuevoPLC1/2.14/1/B	ON=EV Bomba Permeat Tancada
AutoGen_2_15	Punto	NuevoPLC1/2.15/1/B	Detecció Escuma
AutoGen_3_00	Punto	NuevoPLC1/3.0/1/B	ON=Ev Air Membranes Oberta

Figura 11. Exemple de fitxer .cdm de la planta pilot de Castell d'Aro

Com podem observar a l'editor, tenim el dispositiu "NuevoPLC1" que es el PLC amb la configuració corresponent segons el model que hi hagi instal·lat a la planta.

A la columna "Nombre" hi ha l'adreça de memòria per la qual accedim al aparell/sensor i la "Dirección" és l'adreça que ens facilita l'empresa o persona encarregada de la instal·lació i configuració dels aparells al PLC i com a comentari posem el nom de l'aparell per tal d'identificar-lo posteriorment.

Cada vegada que s'instal·la un aparell nou a la planta, per tal de poder accedir al seu valor a través de les nostres aplicacions, s'ha de modificar aquest fitxer de manera que es conegui l'adreça de memòria perquè el servidor OPC la identifiqui i puguem llegir-ne el seu valor.

5.2. El sistema informàtic existent

El punt de partida d'aquest projecte és el sistema que ja hi havia desenvolupat i que estava format per una aplicació en Java anomenada *colmatar*, una aplicació web, una aplicació pel disseny de les regles de control i una base de dades, el detall s'explica als apartats següents.

5.2.1. Aplicació Java *colmatar*

L'aplicació *colmatar* és l'encarregada de rebre les dades a través del servidor OPC i les processa, realitza càlculs si és necessari i ho guarda a la base de dades. Es guarden dades cada 10 segons, de manera que de totes les dades que es reben del PLC disposem dels seus valors cada 10 segons, guardats a la base de dades.

És bàsicament l'aplicació de CONTROL de la planta. A continuació podem veure la informació detallada d'aquesta aplicació.

L'aplicació consta de 8 packages:

- Default package, conté la classe principal.

- Main

- Utilitza les següents llibreries:

```
java.utils.*;  
javax.swing.*;  
java.awt.AWTException;  
java.awt.Image;  
java.awt.MenuItem;  
java.awt.PopupMenu;  
java.awt.SystemTray;  
java.awt.Toolkit;  
java.awt.TrayIcon;  
java.awt.event.ActionEvent;  
java.awt.event.ActionListener;  
java.io.FileOutputStream;  
java.io.IOException;  
java.io.PrintStream;
```

- calcSystem, que conté les classes que s'encarreguen de realitzar els càlculs.

- Calcul

- Utilitza les següents llibreries;

```
java.util.Stack;  
java.util.Vector;  
ru.formula.Formula;
```

- TimerCalc
 - Utilitza les següents llibreries:

```
java.sql.Connection;  
java.sql.ResultSet;  
java.sql.Statement;  
java.text.SimpleDateFormat;  
java.util.Date;  
java.util.Enumeration;  
java.util.Hashtable;  
java.util.Timer;  
java.util.TimerTask;
```
- TimerCalcHour
 - Utilitza les llibreries següents:

```
java.sql.ResultSet;  
java.sql.Statement;  
java.text.DateFormat;  
java.text.SimpleDateFormat;  
java.util.Calendar;  
java.util.Collections;  
java.util.Date;  
java.util.Enumeration;  
java.util.GregorianCalendar;  
java.util.Hashtable;  
java.util.Timer;  
java.util.TimerTask;  
java.util.Vector;  
org.apache.commons.math.stat.StatUtils;  
org.apache.commons.math.stat.descriptive.moment.StandardDe  
viation;  
org.apache.commons.math.stat.regression.SimpleRegression;
```
- caseSystem, que conté caseSystem per solucionar casos.
 - Utilitza les llibreries següents:

```
java.util.Enumeration;  
com.sun.org.apache.xalan.internal.xsltc.runtime.Hashtabl  
e;
```
- Listener, que conté les classes que permeten connexions.

- DssClient
 - Utilitza les llibreries següents:

```
java.io.BufferedReader;  
java.io.InputStreamReader;  
java.io.ObjectInputStream;  
java.io.ObjectOutputStream;  
java.io.PrintWriter;  
java.net.InetAddress;  
java.net.Socket;
```

- DssServer
 - Utilitza les llibreries següents:

```
java.io.*;  
java.util.*;  
java.net.*;
```

- logicSystem, que conté les classes de control.
 - DebugSystem
 - Utilitza les llibreries següents:

```
java.io.File;  
java.io.FileWriter;  
java.io.IOException;
```

 - LogicControl
 - Utilitza les llibreries següents:

```
java.util.Enumeration;  
java.util.Hashtable;
```

 - Timer5min
 - Utilitza les llibreries següents:

```
java.net.UnknownHostException;  
java.text.SimpleDateFormat;  
java.util.Date;  
java.util.Enumeration;  
java.util.Hashtable;  
java.util.Timer;
```



```
java.util.TimerTask;  
javax.swing.JOptionPane;  
org.openscada.opc.lib.common.NotConnectedException;  
org.openscada.opc.lib.da.Async20Access;  
org.openscada.opc.lib.da.DuplicateGroupException;  
org.openscada.opc.lib.da.Server;
```

- o TimerControl

- Utilitza les llibreries següents:

```
java.net.UnknownHostException;  
java.text.SimpleDateFormat;  
java.util.Date;  
java.util.Enumeration;  
java.util.Hashtable;  
java.util.Timer;  
java.util.TimerTask;  
javax.swing.JOptionPane;  
org.jinterop.dcom.common.JIException;  
org.openscada.opc.lib.common.NotConnectedException;  
org.openscada.opc.lib.da.Async20Access;  
org.openscada.opc.lib.da.DuplicateGroupException;  
org.openscada.opc.lib.da.Server;
```

- opcSystem, que conté totes les classes que ens permeten la connexió amb el servidor opc.

- o JavaOPC

- Utilitza les llibreries següents:

```
java.util.*;  
java.util.concurrent.ScheduledExecutorService;  
java.util.logging.Level;  
javax.swing.tree.*;  
javax.swing.JTree;  
org.jinterop.dcom.common.JIException;  
org.jinterop.dcom.common.JISystem;  
org.jinterop.dcom.core.JISession;  
org.jinterop.dcom.core.JIUnsignedInteger;  
org.jinterop.dcom.core.JIVariant;  
org.openscada.opc.lib.common.ConnectionInformation;  
org.openscada.opc.lib.da.AccessBase;
```

```
org.openscada.opc.lib.da.Async20Access;  
org.openscada.opc.lib.da.AutoReconnectController;  
org.openscada.opc.lib.da.AutoReconnectListener;  
org.openscada.opc.lib.da.AutoReconnectState;  
org.openscada.opc.lib.da.Group;  
org.openscada.opc.lib.da.Item;  
org.openscada.opc.lib.da.ItemState;  
org.openscada.opc.lib.da.Server;  
org.openscada.opc.lib.da.SyncAccess;  
org.openscada.opc.lib.da.browser.Branch;  
org.openscada.opc.lib.da.browser.Leaf;  
org.openscada.opc.lib.list.ServerList;
```

- o ListenerASync

- Utilitza les llibreries següents:

```
java.text.SimpleDateFormat;  
java.util.Date;  
org.jinterop.dcom.common.JIException;  
org.jinterop.dcom.core.JIVariant;  
org.openscada.opc.lib.da.DataCallback;  
org.openscada.opc.lib.da.Item;  
org.openscada.opc.lib.da.ItemState;
```

- o ListenerSync

- Utilitza les llibreries següents:

```
java.text.SimpleDateFormat;  
java.util.Date;  
org.jinterop.dcom.common.JIException;  
org.jinterop.dcom.core.JIVariant;  
org.openscada.opc.lib.da.DataCallback;  
org.openscada.opc.lib.da.Item;  
org.openscada.opc.lib.da.ItemState;
```

- o ListenerSync30

- Utilitza les llibreries següents:

```
java.text.SimpleDateFormat;  
java.util.Date;  
org.jinterop.dcom.core.JIVariant;  
org.openscada.opc.lib.da.DataCallback;
```

```
org.opencada.opc.lib.da.Item;  
org.opencada.opc.lib.da.ItemState;
```

- o ListenerSync5min

- Utilitza les llibreries següents:

```
java.text.SimpleDateFormat;  
java.util.Date;  
org.jinterop.dcom.common.JIException;  
org.jinterop.dcom.core.JIVariant;  
org.opencada.opc.lib.da.DataCallback;  
org.opencada.opc.lib.da.Item;  
org.opencada.opc.lib.da.ItemState;
```

- o TimerData

- Utilitza les llibreries següents:

```
java.net.UnknownHostException;  
java.text.SimpleDateFormat;  
java.util.Date;  
java.util.Enumeration;  
java.util.Hashtable;  
java.util.Timer;  
java.util.TimerTask;  
javax.swing.JOptionPane;  
org.opencada.opc.lib.common.NotConnectedException;  
org.opencada.opc.lib.da.Async20Access;  
org.opencada.opc.lib.da.DuplicateGroupException;  
org.opencada.opc.lib.da.Server;
```

- persistenceSystem, que conté les classes que ens permeten la connexió amb la base de dades.

- o Connexio

- Utilitza les llibreries següents:

```
java.sql.Connection;  
java.sql.DriverManager;  
java.sql.SQLException;  
java.util.Hashtable;  
java.util.Vector;
```

- MySQLPersistence
 - Utilitza les llibreries següents:

```
java.sql.Connection;  
java.sql.PreparedStatement;  
java.sql.ResultSet;  
java.sql.SQLException;  
java.sql.Statement;  
java.text.DateFormat;  
java.text.SimpleDateFormat;  
java.util.Calendar;  
java.util.Collections;  
java.util.Date;  
java.util.Enumeration;  
java.util.Hashtable;  
java.util.Vector;  
org.apache.commons.math.stat.StatUtils;  
org.apache.commons.math.stat.descriptive.moment.StandardDe  
viation;  
org.apache.commons.math.stat.regression.SimpleRegression;
```
- MySQLPersistenceHistoValor
 - Utilitza les llibreries següents:

```
java.sql.Connection;  
java.sql.ResultSet;  
java.sql.SQLException;  
java.sql.Statement;  
java.text.DateFormat;  
java.text.SimpleDateFormat;  
java.util.Calendar;  
java.util.Collections;  
java.util.Date;  
java.util.Vector;  
javax.swing.JProgressBar;  
org.apache.commons.math.stat.StatUtils;  
org.apache.commons.math.stat.descriptive.moment.StandardDe  
viation;  
org.apache.commons.math.stat.regression.SimpleRegression;
```
- ThreadPersistenceMySQL

- Utilitza les llibreries següents:
 - `java.text.DateFormat;`
 - `java.text.SimpleDateFormat;`
 - `java.util.Date;`
 - `java.util.Enumeration;`
 - `java.util.GregorianCalendar;`
 - `java.util.Hashtable;`
 - `java.util.Timer;`
 - `java.util.TimerTask;`
 - `java.util.Vector;`
- ThreadPersistenceMySQL10sec
 - Utilitza les llibreries següents:
 - `java.text.DateFormat;`
 - `java.text.ParseException;`
 - `java.text.SimpleDateFormat;`
 - `java.util.Calendar;`
 - `java.util.Date;`
 - `java.util.Enumeration;`
 - `java.util.GregorianCalendar;`
 - `java.util.Hashtable;`
 - `java.util.Timer;`
 - `java.util.TimerTask;`
 - `java.util.Vector;`
- Utils, que conté les classes utilitàries.
 - Observer
 - Patró Observer
 - Rellotge
 - Utilitza les llibreries següents:
 - `java.text.SimpleDateFormat;`
 - `java.util.Date;`
 - `java.util.Timer;`
 - `java.util.TimerTask;`
 - `javax.swing.JLabel;`
 - `javax.swing.JOptionPane;`
 - `java.util.Calendar;`

```
java.util.GregorianCalendar;
```

- o Utils

- Utilitza les llibreries següents:

```
java.io.IOException;  
java.io.StringReader;  
java.util.Enumeration;  
java.util.Hashtable;  
java.util.Iterator;  
java.util.List;  
java.util.Vector;  
org.jdom.Document;  
org.jdom.Element;  
org.jdom.JDOMException;  
org.jdom.filter.ElementFilter;  
org.jdom.input.SAXBuilder;  
org.jdom.output.XMLOutputter;  
org.xml.sax.InputSource;
```

- o Value

- Utilitza les llibreries següents:

```
java.util.*;
```

5.2.2. Aplicació web

L'aplicació web és la que s'encarrega d'interactuar amb l'usuari, de manera que es pot visualitzar l'esquema de la planta, una imatge en directe de la planta, les dades dels diferents aparells/sensors de què disposa la planta, així com els seus gràfics, les dades històriques mitjançant informes o també alguns gràfics, etc.

A la següent imatge es pot veure l'aplicació web que es detalla a continuació.

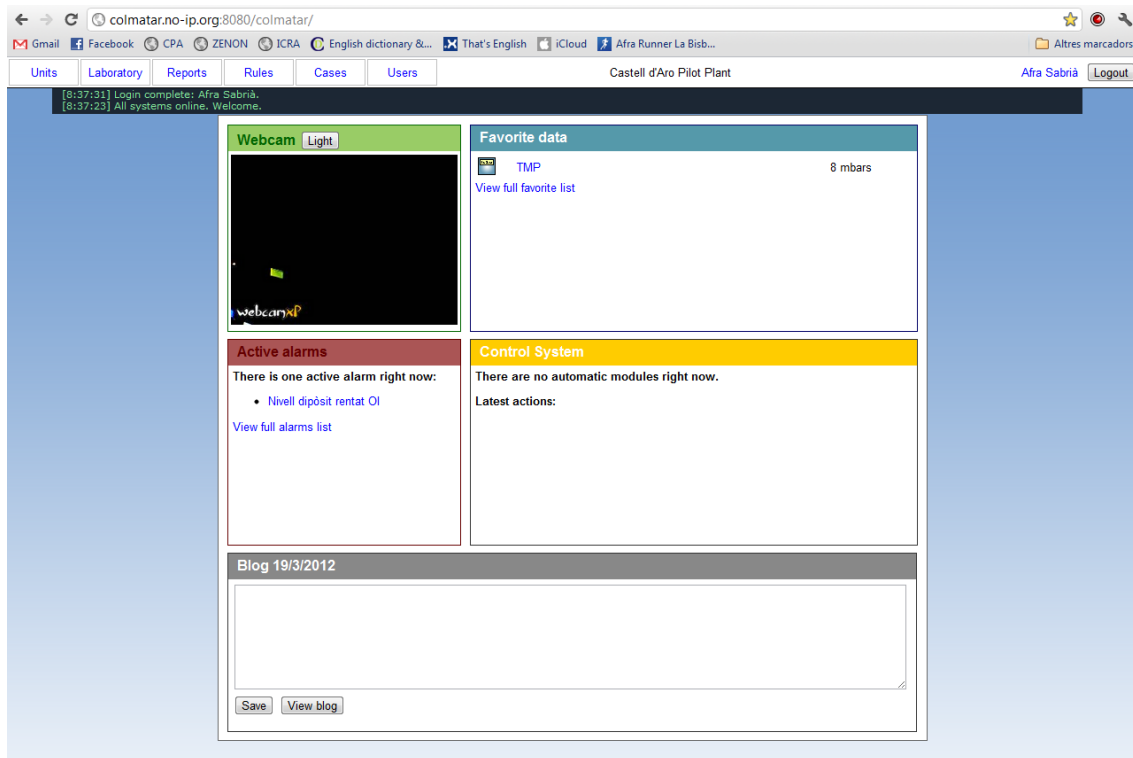


Figura 12. Vista de l'aplicació web

Informació detallada de l'aplicació web:

Fitxers js:

- Autenticació
- Blog
- Cálculs
- Calendar
- Casos
- Components
- Favorits
- infoBar
- informes
- interfície

- mesures
- panell
- precarregalmatges
- unitats
- usuaris

Fitxers jsp:

- servidor_casos
- servidor_element
- servidor_formula
- servidor_general
- servidor_informes
- servidor_login
- servidor_mesures
- servidor_normes
- servidor_usuaris

Fitxer html:

- index

Fitxer css:

- index

Fitxer XML:

- configuracio

A la següent imatge podem veure la pantalla d'inici que ens permet veure l'esquema de la planta, els aparells que hi ha i poder identificar un usuari, per tal d'accedir a l'aplicació.

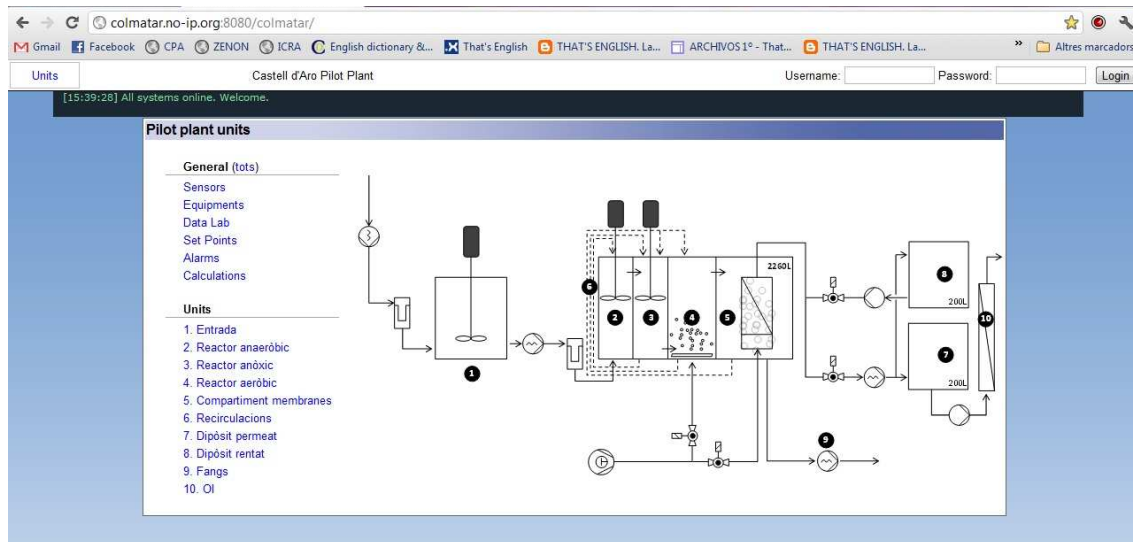


Figura 13. Pantalla principal

A continuació, a la Figura 14, podem veure un cop identificat un usuari les opcions que té, segons els seus privilegis. La imatge ha sigut obtinguda amb un usuari amb privilegis de Super Administrador.

Tal i com es pot veure a la imatge de la Figura 14 hi ha varis panells. A dalt, a l'esquerra sota les pestanyes, podem veure la imatge en directe de la planta pilot mitjançant una webcam. Al costat podem veure *Favorite data* on cada usuari posarà les dades que vol veure quan iniciï sessió. A baix, a l'esquerra tenim *Active Alarms* que ens indica les alarmes que hi ha actives en aquell moment i al seu costat el panell *Control System* que un cop implementat el nou mòdul de regles de control, ens indicarà les que hi ha actives en aquell moment. A sota de tot podem veure el panell *Blog* on els usuaris podran anotar aspectes que els semblin rellevants pel funcionament de la planta, o bé canvis que es realitzin.

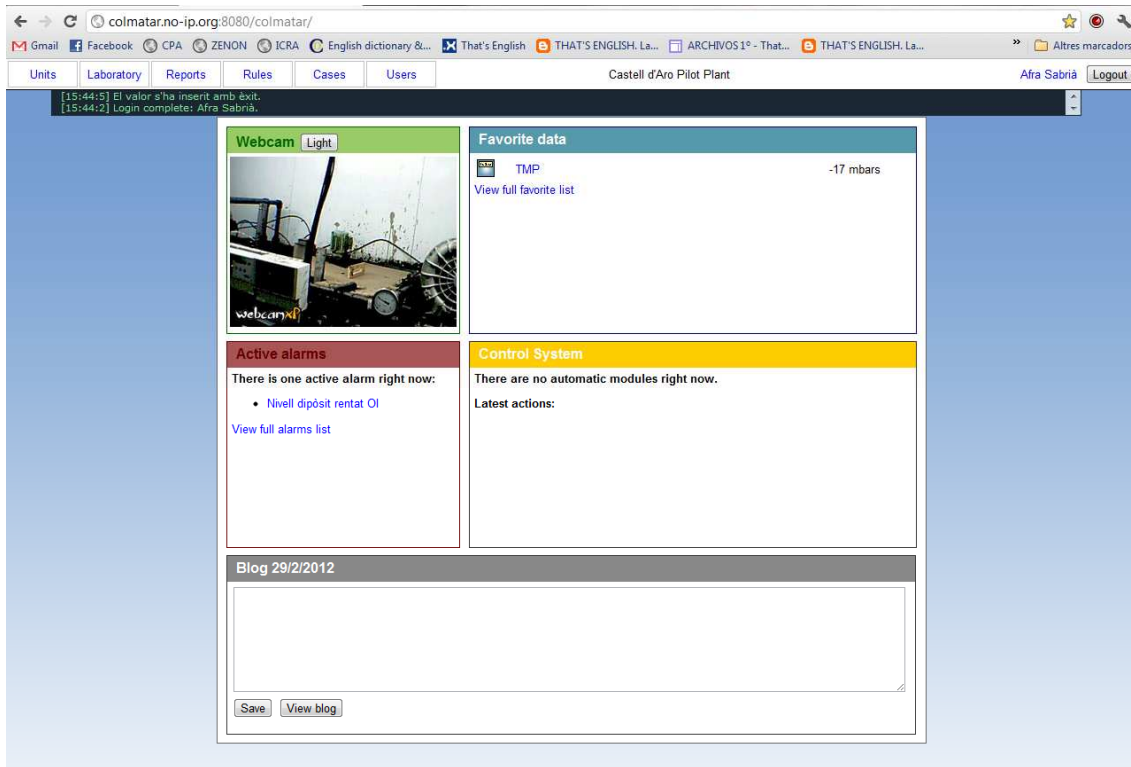


Figura 14. Inici amb usuari

A continuació es descriu amb més detall cada pestanya que veiem a la pàgina principal.

La pestanya *Units* ens porta a la pàgina principal.

Des d'allà podem seleccionar algun aparell o unitat de la planta de la qual vulguem visualitzar-ne el valor i també es visualitza el gràfic de les dades de les últimes vint-i-quatre hores, com es veu a la imatge següent:

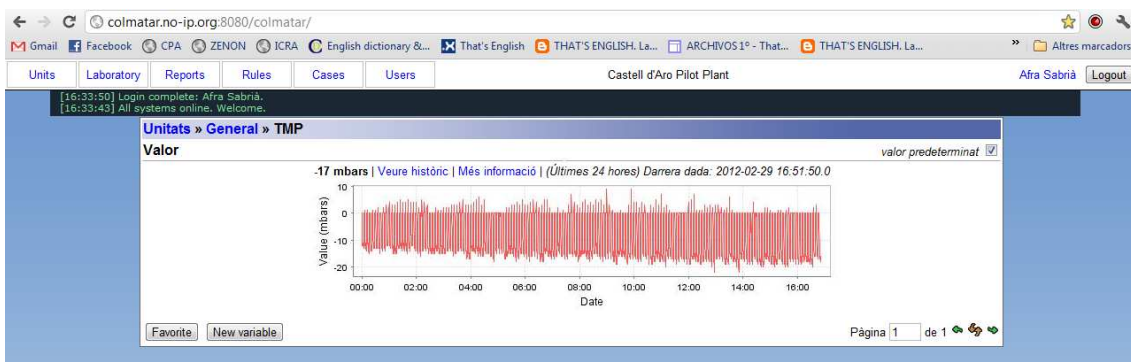


Figura 15. Exemple de visualització d'un aparell

La pestanya *Laboratory* permet entrar dades de Laboratori manualment per tal que quedin guardades a la base de dades i poder consultar els històrics, com es veu en la imatge següent.

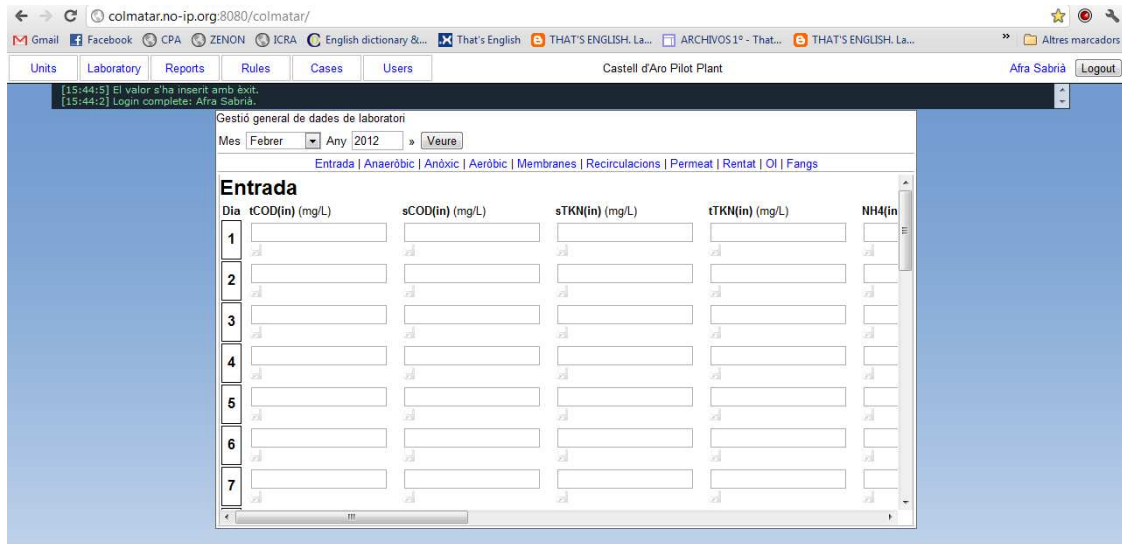


Figura 16. Visualització pestanya Laboratory

La pestanya Reports ens permet descarregar informes, cada informe amb les seves particularitats i demanant les dades d'entrada que siguin necessàries.

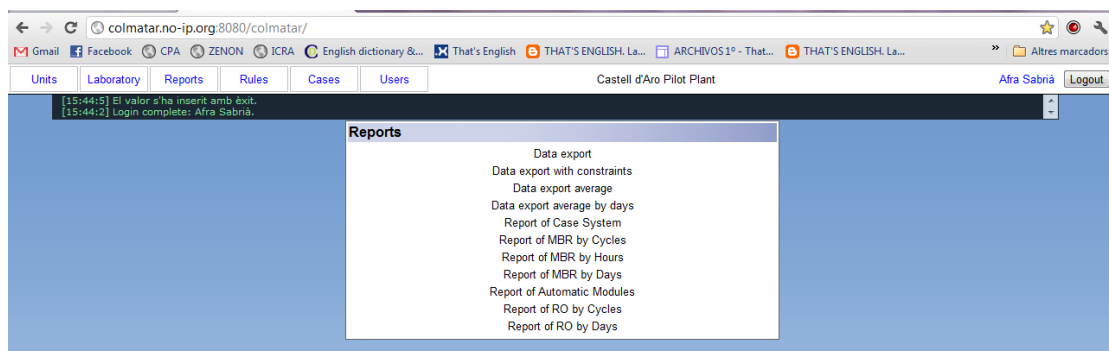


Figura 17. Visualització pestanya Reports

La pestanya *Rules* a desenvolupar en aquest projecte, és on es visualitzaran les regles de control que hi hagi implementades, si estan actives o no, i el resultat de la seva última execució.

La pestanya *Cases* per crear casos.

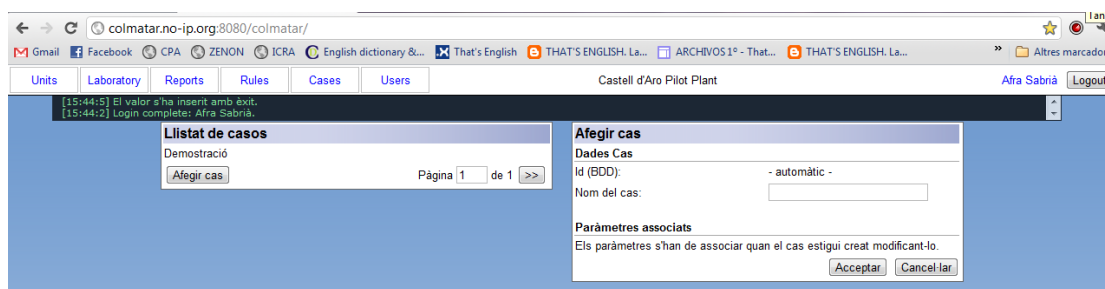


Figura 18. Visualització pestanya Casos

I per últim la pestanya *Users* on podem visualitzar els usuaris que hi ha donats d'alta a l'aplicació i segons els privilegis, també poder donar d'alta nous usuaris.

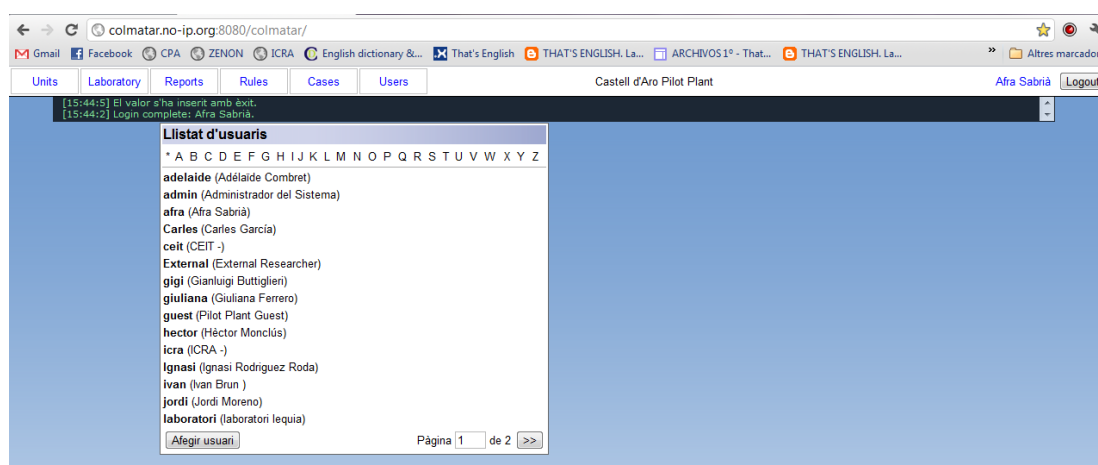


Figura 19. Visualització pestanya Users

5.2.3. Aplicació pel disseny de les regles de control.

Aquesta aplicació va ser desenvolupada per un antic membre del grup LEQUIA i que serveix per poder dissenyar de manera més fàcil les regles de control a implementar.

De manera que l'informàtic (en aquest cas jo) o bé un usuari expert que tingui coneixement de l'aplicació pugui mitjançant l'esquema o arbre de decisions que se li facilita, dissenyar una regla de control, tenint en compte els inputs que necessita, les variables de control i les sortides necessàries, tot això tenint en compte si els inputs necessaris ens venen donats directament per un valor del PLC o bé cal implementar uns nous càlculs.

Un cop tenim dissenyada la regla de control, es crea un fitxer XML que conté tota la informació de la regla de control i que és el que l'aplicació Java *colmatar* interpretarà per tal d'executar-la automàticament.

La imatge de la Figura 20 que podem veure a continuació mostra la pantalla de l'aplicació de disseny de les regles de control. On crearem una nova regla de control a la *Lista de arboles*. A "Entrada" hi posarem les variables d'entrada necessàries pel control, a *Salida* les variables de sortida i a *Control* variables de control que ens puguin ser útils a l'hora de dissenyar la regla de control.

A la pestanya anomenada *no Grupo* i que hi podem posar nom, s'aniran creant les diferents branques o diferents condicions que ens siguin necessàries pel desenvolupament de la regla de control, així com les condicions d'entrada dins el requadre *Condiciones de entrada de la regla* i les seves corresponents conseqüències a *Consecuencia de la regla*.

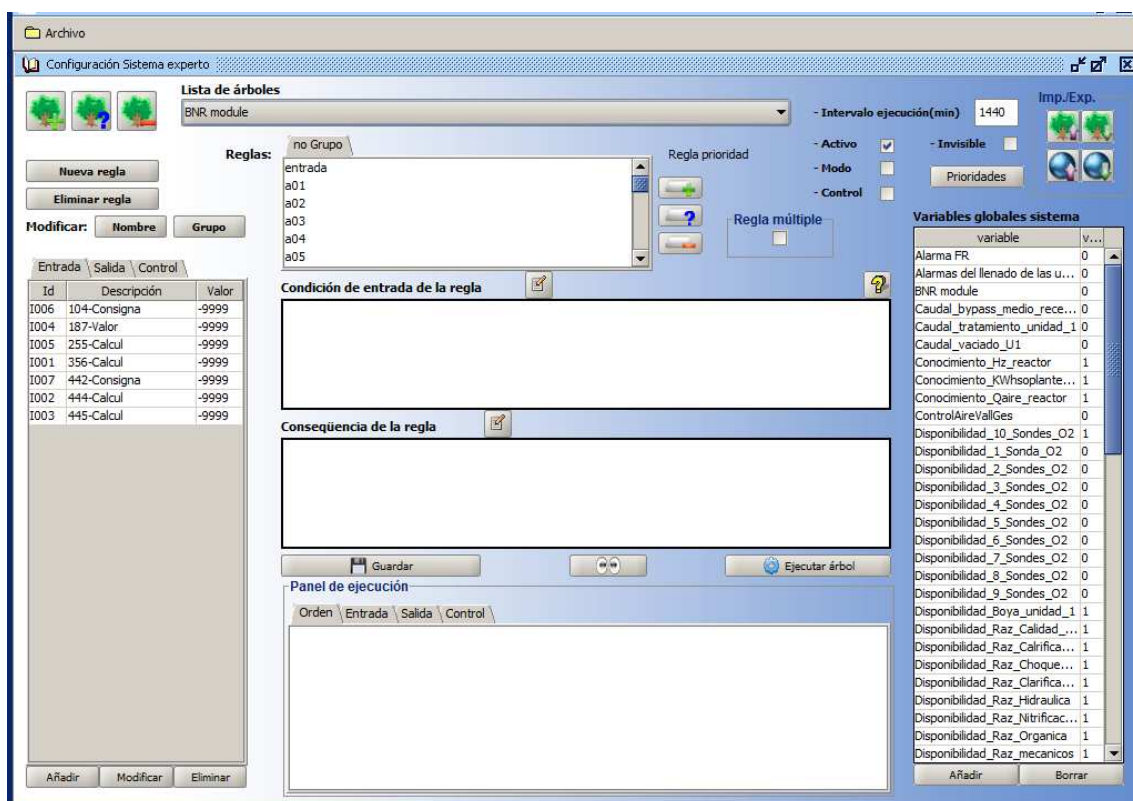


Figura 20. Editor de reglas de control

Aquesta aplicació utilitza una base de dades postgresQL, que és on quedarà guardada la regla de control dissenyada en forma de fitxer XML i d'on el podem extreure per tal de poder-la utilitzar dins l'aplicació "colmatar".

5.2.4. Base de dades

Utilitzem una base de dades MySQL ja existent i que conté les taules que es detallen a continuació.

Taules:

- aparells: hi guardem tots els aparells de què disposem així com el seu tipus (Alarma, Consigna, Sensor, etc.)
- blog: es guarden totes les anotacions fetes al blog amb la seva data.
- casos: s'hi guarden els casos creats.
- casos variables: s'hi guarden les variables utilitzades en els casos creats.
- episodi: s'hi guarden les execucions dels casos.
- favorits: es guarden els aparells favorits de cada usuari (es visualitzen a l'aplicació web, al identificar-se l'usuari)
- flags: es guarden marcadors.
- formules: es guarden formules que necessiten algunes variables per tal de visualitzar-les com és necessari.
- histolab: s'hi guarda les dades de laboratori introduïdes.
- histovalor5min: les dades dels aparells cada 5 minuts.
- histovalor: anteriorment s'hi guardaven totes les dades, aquesta taula va créixer massa i actualment es guarda una taula per cada mes de cada any.
- histovalor_2XXX_XX (tantes taules com mesos per any hi hagi, per exemple histovalor_2012_01, seria la taula que conté les dades del mes de gener del 2012): s'hi guarden les dades de tots els aparells cada 10 segons.
- item: conté tots els aparells de què disposem juntament amb la seva corresponent adreça de memòria, les unitats i el rang de valors que pot adquirir.
- opc item
- sistema opc: guarda la configuració necessària per a la connexió amb el servidor OPC.
- tmvalor: guarda les dades de cada aparell de les últimes 24h.

- trvalor: guarda la última dada rebuda de cada aparell.
- unitats
- usuaris: guarda tots els usuaris que poden iniciar sessió a l'aplicació web i els seus privilegis.
- write_items: taula utilitzada de "pont" per quan hi ha algun canvi de valor d'un aparell, es canvia des de l'aplicació web i està a aquesta taula fins que es traspasa i es fa el canvi de valor al PLC.

5.3. Grup MBR del LEQUIA

Actualment el grup de MBR del LEQUIA està format pels membres següents:

- Dr. Ignasi Rodriguez-Roda (Investigador Principal)
- Dr. Joaquim Comas (Investigador Principal)
- Dr. August Bonmatí
- Dr. Gianluigi Buttiglieri
- Dra. Giuliana Ferrero
- Dr. Hèctor Monclús
- Neus Collado (PhD Student)
- Montse Dalmau (PhD Student)
- Sara Gabarrón (PhD Student)
- Jordi Moreno (PhD Student)
- Ivan Brun (PhD Student)
- Mariona Casadevall (PhD Student)
- Maria Casellas (PhD Student)
- Dr. Franz Wotawa (Tu Graz - External Researcher)

- Dr. Jean Philippe Steyer (INRA - External Researcher)
- Afra Sabrià (Software Design)

Les meves tasques dins del grup són bàsicament tot el que engloba la part informàtica del projecte, des de la preparació de les màquines en quan a sistema operatiu, software, configuració, etc... fins a l'anàlisi i desenvolupament de nous mòduls, aplicacions o modificacions que siguin necessàries per al projecte COLMATAR+.

També he de vetllar perquè tot el sistema informàtic que hi ha muntat a cada planta pilot (com he esmentat abans, n'hi ha varies) funcioni correctament, solucionar possibles problemes, còpies de seguretat, etc.

Aquest projecte final de carrera que es presenta ha donat suport informàtic a dues Tesis de membres del grup LEQUIA.

- *Development of a decision support system for the integrated control of membrane bioreactors* del Dr. Hèctor Monclús Sales (2011).
- *Development of an air-scour control system for membrane bioreactors* de la Dra. Giuliana Ferrero (2011).

5.4. Què es vol fer?

Una vegada explicat el sistema informàtic existent anem a comentar què es vol fer concretament en aquest projecte final de carrera.

Disposem d'una aplicació desenvolupada amb Java que s'encarrega de recollir i processar les dades que rebem del PLC a través del servidor OPC i les guarda a la Base de dades MySQL. Les dades les podem visualitzar a través d'una aplicació web desenvolupada amb JavaScript i AJAX.

Es vol afegir un nou mòdul que mitjançant uns paràmetres ens permeti realitzar canvis a la planta de manera automàtica sense que hi hagi d'intervenir cap usuari.

El mòdul de control automàtic ja estava planificat com desenvolupar-lo, de manera que disposem d'una petita aplicació que podria utilitzar un usuari expert per tal de poder dissenyar una nova regla de control.

S'ha de veure com a partir del fitxer XML que crea aquesta aplicació, la nostra aplicació d'adquisició de dades en Java ho pot processar, fer-ne l'execució automàtica i guardar-ne les dades resultants.

S'haurà d'afegir també un nou mòdul que permeti enviar correus electrònics informant de les alarmes actives de la planta pilot.

S'ha de tenir en comptes quines dades tenim i quines necessitem, per tant sabem que disposem d'una aplicació ja desenvolupada per un membre del grup Lequia on podem implementar l'arbre de decisions transformant-lo a un document XML.

Necessitem desenvolupar un nou mòdul a l'aplicació d'adquisició de dades que ens permeti llegir i interpretar aquest XML, executar l'arbre de decisions dissenyat en un moment determinat i que les dades resultants d'aquesta execució quedin guardades per tal de poder visualitzar els canvis que s'han realitzat.

La visualització d'aquestes dades i el control de si s'ha d'executar alguna regla automàtica es realitzarà a través de l'aplicació web que és la que té interacció amb l'usuari.

Aquesta aplicació ja desenvolupada ens permet separar els aparells dels que llegim les dades o modifiquem, en tres grups: input, control i output.

Caldrà identificar quines operacions i càlculs cal fer. El mòdul d'automatitzacions en si, no requereix cap operació o càlcul previ ja que es tracta d'interpretar un format determinat de fitxer i executar el que sigui necessari.

On hi ha necessitat de realitzar operacions o càlculs és dins de cada regla de control o arbre de decisions que dissenyi l'usuari expert. I dependrà del tipus d'accions que es vulguin realitzar.

Segons els requisits d'operacions o càlculs que necessitem per a l'execució de la regla de control, s'haurà d'afegir o modificar codi a l'aplicació "colmatar" que és la que ens permet tractar les dades que rebem directament des del PLC.

Així mateix per l'enviament de correus electrònics també s'haurà de modificar dins el codi les adreces electròniques on s'enviaran els correus.

Les proves de validació de les operacions i càlculs bàsicament són comprovar que els càlculs es realitzin correctament segons els criteris que s'hagin decidit i que es guardi correctament també a la base de dades.

Un cop tenim el mòdul a l'aplicació "colmatar" que ens permeti llegir, interpretar i guardar les execucions, cada regla de control que s'hagi d'afegir al control, s'haurà de dissenyar i crear al software que crea el fitxer XML.

Els jocs de proves es dissenyaran segons cada regla de control creada, per tal de valorar segons l'arbre de decisions que tinguem, que fa les accions correctes.

Les proves de validació del mòdul primerament es fan en servidor local, simulant els valors dels aparells i comprovant que realment s'efectuen les accions definides. Un cop hem realitzat aquestes proves de validació i són correctes, ho posem en funcionament a escala real, és a dir, a planta pilot on els responsables valoraran segons les execucions i l'estat de la planta, si la regla de control està actuant com s'esperava.

6. Requisits del sistema

Els requisits de l'aplicació a desenvolupar en aquest PFC són:

- El sistema ha de permetre recollir les dades necessàries per a realitzar correctament el control automàtic, ja sigui adquirint directament les dades del PLC, o bé de la base de dades.
- Podrem adquirir les dades directament des del PLC si la dada que ens dona l'aparell és la que necessitem per poder executar el control i no és necessari fer-li cap tractament. Adquirirem les dades des de la Base de dades en el cas que les dades que necessitem no ens vinguin directament donades pel PLC i sigui necessari realitzar alguna transformació, operació o càlcul amb les dades que ens venen directament des del PLC i un cop realitzat quedi guardat a la base de dades.
- Segons les dades d'entrada fer les actuacions pertinents descrites a cada "regla de control" que hi hagi activa dins el mòdul de controls automàtics. Les actuacions que hagi de realitzar cada regla de control vénen descrites en la implementació de l'arbre de decisions dissenyat i guardat en format XML.
- Modificar el/els valors que siguin necessaris segons hagi decidit el sistema de control i guardar la informació necessària per després poder obtenir-ne un històric de les actuacions. En la modificació dels valors també es realitza el canvi del valor donant ordre cap al PLC, d'aquesta manera canviar l'estat/s o valor/s que es decideixen en el sistema de control.
- Les actuacions que hagi fet el sistema de control s'han de poder visualitzar a través de l'aplicació web, ja sigui específicament comprovant l'ítem o aparell/s que intervenen en cada "regla de control", mitjançant l'apartat del mòdul de control automàtic on es mostraran les últimes accions realitzades i les dades necessàries decidides anteriorment o bé mitjançant un informe on es podrà veure totes les "variables" que hi han intervingut.
- El sistema també ha de permetre enviar correus electrònics als responsables de la planta pilot quan hi ha la detecció d'alguna alarma, per tal de poder solucionar el problema el més aviat possible i que no afecti al bon funcionament de la planta.

- També caldrà modificar l'aplicació web per tal de poder visualitzar les execucions de les regles de control i les accions realitzades.

7. Estudis i decisions

Una vegada explicat el marc de treball existent i analitzats els requisits principals del nou mòdul a desenvolupar i per tal de poder dur a terme l'objectiu principal del projecte que és un sistema d'ajuda a les decisions intel·ligent, caldrà desenvolupar un nou mòdul dins l'aplicació informàtica en Java encarregada d'adquirir, manipular i guardar les dades que provenen del PLC de la planta.

També caldrà dissenyar com guardem les dades de les actuacions del sistema de control dins la nostra base de dades ja existent.

Decidirem com desenvolupem cada "regla de control" per tal que sigui fàcil d'executar automàticament.

Modificarem o afegirem codi nou per tal de poder visualitzar el llistat de regles de control, les que hi ha actives actualment, els resultats de les últimes actuacions, etc.

Caldrà afegir un nou mòdul per tal de poder enviar correus electrònics als responsables de la planta pilot, per tal que s'assabentin de les alarmes que hi hagi actives en un moment determinat i poder solucionar els problemes per tal que no afecti al bon funcionament de la planta.

Disposem d'un PLC-SCADA que ens llegeix totes les dades dels aparells que hi ha instal·lats a la planta pilot, a través d'un servidor OPC

Disposem d'una aplicació en llenguatge Java, anomenada "colmatar" que s'encarrega de llegir les dades que provenen del PLC a través d'un servidor OPC.

L'aplicació Java tal i com s'explica a l'apartat 5 utilitza unes llibreries per al seu funcionament, a l'afegir el nou mòdul veurem quines llibreries haurem d'afegir.

En aquesta aplicació anomenada "colmatar" és on s'haurà de desenvolupar un nou mòdul que permeti llegir i executar les regles de control i guardar-ne els canvis que puguin haver-hi. També en aquesta mateixa aplicació és on afegirem un nou mòdul per tal de enviar correus electrònics que informin de les alarmes actives.

S'haurà de decidir com es guardaran les execucions de les regles de control a la base de dades, per tal de poder consultar les execucions pertinents i les seves accions.

Per altra banda disposem també d'una aplicació en llenguatge Java que ens permet de manera visual i senzilla de dissenyar l'arbre de decisions o la regla de control per tal de transformar-la en un fitxer XML que l'aplicació colmatar s'encarregarà de interpretar i executar.

Aquesta aplicació utilitza una base de dades postgresSQL per tal d'emmagatzemar les regles de control dissenyades.

Per últim disposem d'una aplicació web que és la que utilitza l'usuari per tal d'interactuar amb la planta pilot, ja sigui visualitzant aparells, gràfiques, descarregar-se informes, visualitzar alarmes, etc. com per modificar valors d'aparells o per poder visualitzar les regles de control que hi ha actives i els canvis que s'hagin pogut realitzar.

8. Anàlisi i disseny del sistema

Al disposar de varies aplicacions que estan relacionades entre si, caldrà analitzar de manera detallada les necessitats del sistema, tenint en compte cada aplicació i dissenyar una solució per al sistema que s'adapti a les necessitats requerides.

8.1. Aplicació Colmatar

En aquest apartat es descriurà amb detall l'aplicació Colmatar.

8.1.1. Anàlisi de com està estructurada

Aquesta aplicació es comunica bàsicament amb la base de dades i amb el servidor OPC.

Partirem d'un document XML que conté totes les condicions i conseqüències d'execució de la regla de control i que estarà guardat a la base de dades.

Per tant s'haurà de dissenyar una classe per a la interpretació d'aquest XML i la manera en què es guardaran les dades que conté, les seves prioritats, com s'executarà, cada quan temps s'executarà, com i on es guardarà la informació de l'execució i els canvis a realitzar, tal i com es pot veure a la Figura 21.

Per altra banda també necessitarem implementar un sistema que permeti enviar correus electrònics a destinataris predeterminats per tal d'informar les alarmes que en aquell moment hi ha actives a la planta pilot, com es veu a la Figura 22.

Normalment als arbres de decisions que l'usuari dissenyi per a crear la regla de control, possiblement caldrà realitzar càlculs diaris que no ens vindran donats per les dades recollides del PLC. És a dir, que s'hauran de fer càlculs de 24 hores que necessitarem per a la implementació de la majoria de regles de control.

A continuació podem veure els diferents models de processos que es realitzaran a l'aplicació Java colmatar.

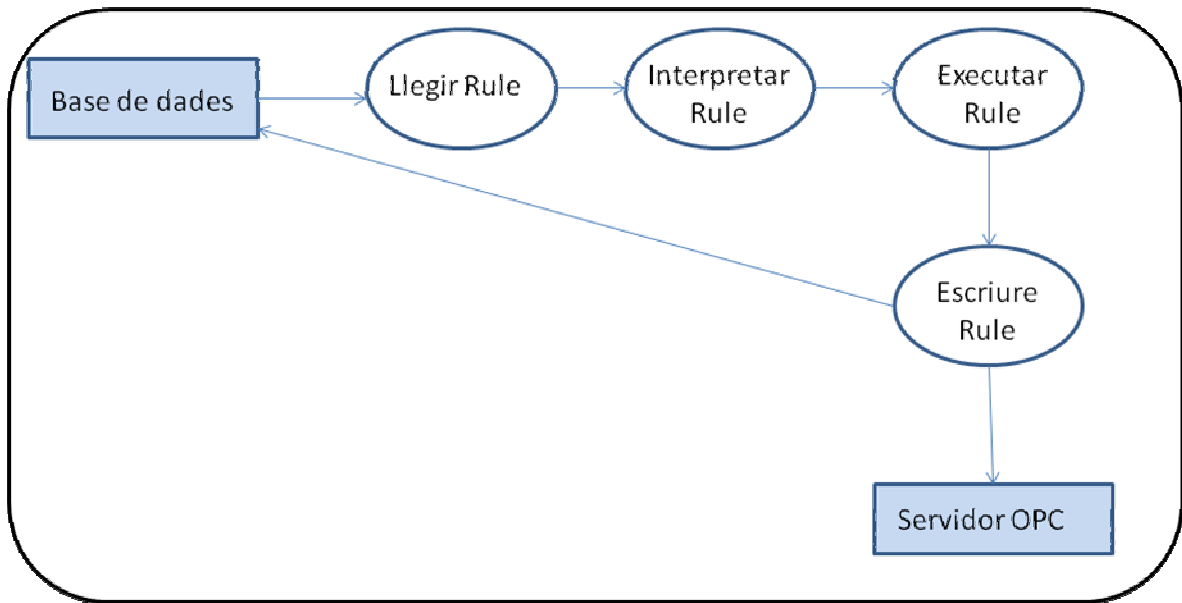


Figura 21. Model de processos de la regla de control

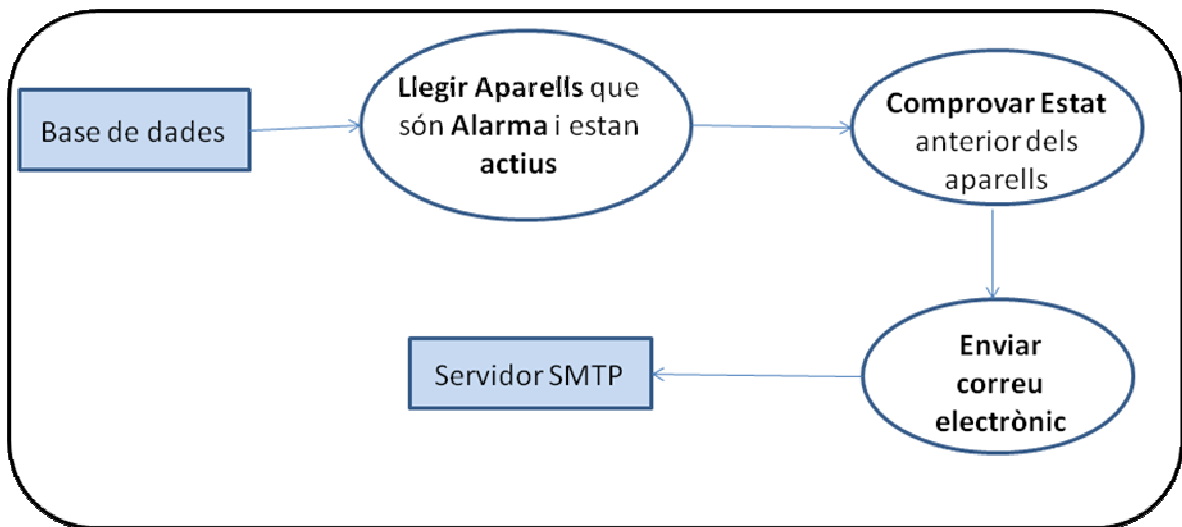


Figura 22. Model de processos d'enviament de correus electrònics

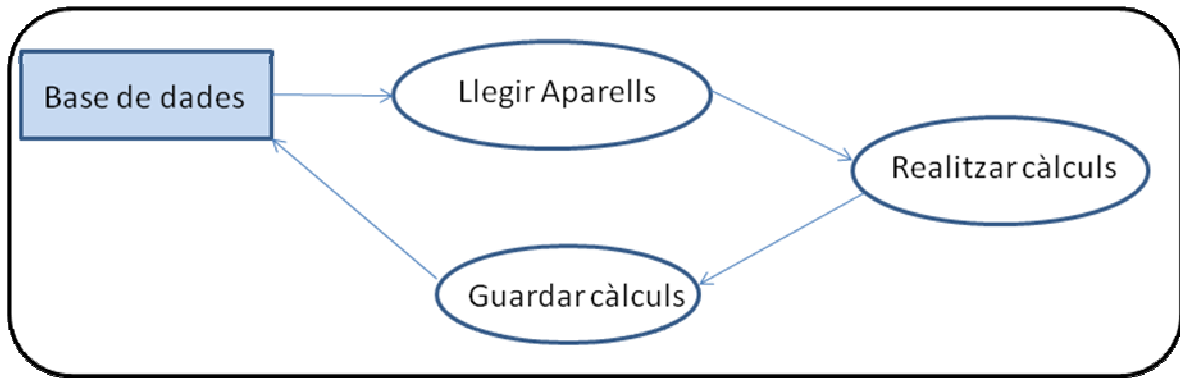


Figura 23. Model de processos de càlculs diaris

Per al model de dades, com que haurem de guardar un document XML a la base de dades que és el que conté tota la informació de la o les regles de control dissenyades, necessitarem una taula que ens permeti tenir totes les regles de control de què disposem i si estan actives o no.

També ens caldrà guardar la informació d'execució de cadascuna d'elles, per tal de saber primerament quina regla de control s'ha executat, en quina data i en quina hora, i també necessitarem saber els aparells que hi intervenen, els seus valors i els canvis que es realitzen.

8.1.2. Disseny

Per a l'enviament de correus electrònics s'afegirà un nou package a l'aplicació que contindrà dues classes.

- mailSystem, que ens permetrà enviar correus electrònics.
- EnviaMail: en aquesta classe hi ha guardada tota la informació referent a la configuració del servidor de correu electrònic utilitzat (en aquest cas Gmail) i també les adreces de correu a qui s'enviaran els avisos d'alarma.
- TimerMail: en aquesta classe hi ha programada una tasca de manera que cada 10 minuts s'accedeix a la base de dades i es comproven els últims valors dels aparells de tipus "Alarma" i que estan actius (taula tvalor). Disposarem d'un vector on es guardaran les alarmes que hi ha actives de manera que cada vegada que es comprovin les alarmes es compararà amb el vector, si les alarmes actives són les mateixes que hi havia en la última comprovació no s'enviarà el correu electrònic, en canvi si hi ha hagut algun canvi ja sigui per una nova alarma activa o bé alguna que ja s'hagi desactivat s'enviarà el correu electrònic informant de les alarmes actives en aquell moment.

Ja que necessitem per a la implementació de la majoria de regles de control alguns càlculs diaris, haurem d'afegir una nova classe al package *calcSystem*.

- TimerCalcDay: en aquesta classe hi ha programada una tasca que s'executarà cada vint-i-quatre hores (a les dotze de la nit). A través d'uns altres càlculs que es realitzen a la classe *MySQLPersistenceHistoValor* del package *persistenceSystem* que ens permetran calcular de cada aparell que prèviament sabem que en necessitem la seva mitjana diària. I des d'aquesta mateixa classe realitzar el càlcul pertinent i guardar els valors calculats a la base de dades.

Per a la interpretació i execució de les regles de control hem desenvolupat diverses classes que hem distribuït en diferents packages.

Distribució per packages:

- *ruleSystem*, ens permetrà inicialitzar el que necessitem.
 - RuleBase: ens permet inicialitzar tot el que necessitem, connexió a la base de dades, aparells que utilitzem, el timer per l'execució, etc.
- *ruleSystem.loader*, tot el relacionat amb l'execució de la regla de control.
 - RuleInfo: declaració dels paràmetres que utilitzarem per guardar dades de la regla de control.
 - RuleLoader: ens permet executar la regla de control a partir del fitxer XML.
 - TimerLoader: ens permet crear una tasca per tal d'executar les regles de control actives cada cert temps.
- *ruleSystem.persistence*, ens permetrà guardar informació.
 - MySQLRulePersistence: ens permet l'execució de la regla de control, guardar les dades de l'execució a la base de dades i tota la informació relacionada.
- *ruleSystem.tree*, ens permetrà interpretar la regla de control.
 - Cause: causes de l'arbre descrit al fitxer XML.
 - Consequence: conseqüències de l'arbre descrit al fitxer XML.

- **Parameter:** paràmetres a utilitzar.
 - **Rule:** relacionat amb la regla de control, nom, causes, conseqüències, prioritats, etc.
 - **Tree:** ens permet obtenir, input, control i paràmetres relacionats amb l'arbre a partir del fitxer XML.
 - **TreeControl:** arbre relacionat amb els aparells de control.
 - **TreeInput:** arbre relacionat amb els aparells d'entrada, que ens permet llegir els seus valors.
 - **TreeOutput:** arbre relacionat amb els aparells de sortida, que ens permet fer canvis en els aparells de sortida.
 - **TreeSystem:** ens permet guardar les variables utilitzades en l'arbre.
- **ruleSystem.xml,** ens permetrà estructurar el fitxer XML.
 - **XMLRuleParser:** ens permet estructurar tot el descrit en el fitxer XML, de manera que sigui més fàcil a l'hora de treballar-hi.

8.1.3. Diagrama de dependències

En el diagrama següent podem veure marcades les parts en les que s'ha treballat en aquest PFC.

Marcats en vermell podem veure la part de *mailSystem* per a l'enviament de correus electrònics quan hi ha Alarmes, també tota la part de *ruleSystem* per al desenvolupament del mòdul de control automàtic que ens permetrà llegir les regles de control dissenyades i en blau la part on s'hi ha fet modificacions a *persistenceSystem* per al càlculs.

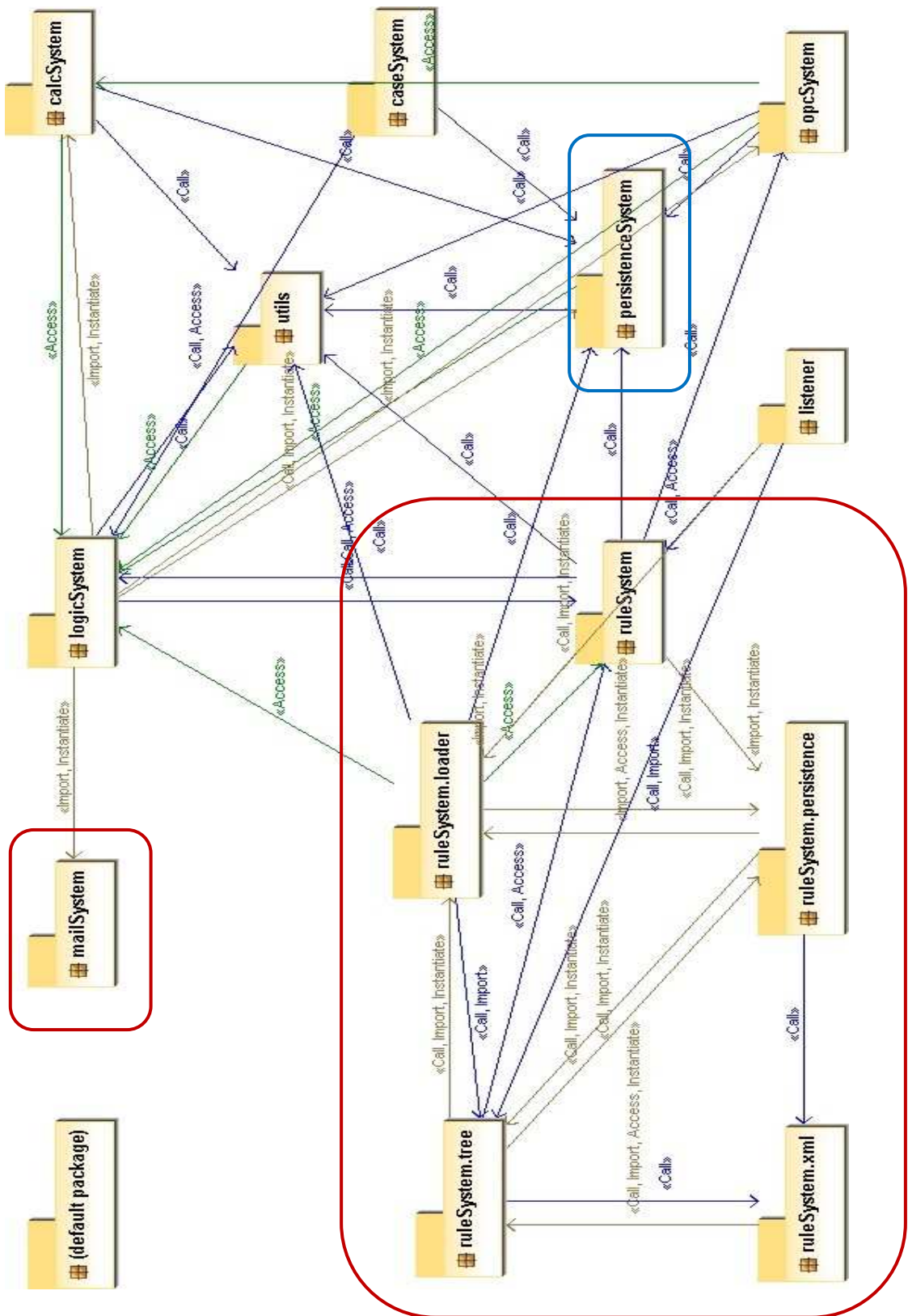


Figura 24. Diagrama de dependències

8.1.4. Diagrames de classes

A continuació descriurem cadascuna de les classes que formen l'aplicació *colmatar*.

calcSystem: conté les classes que s'encarreguen de realitzar càlculs.

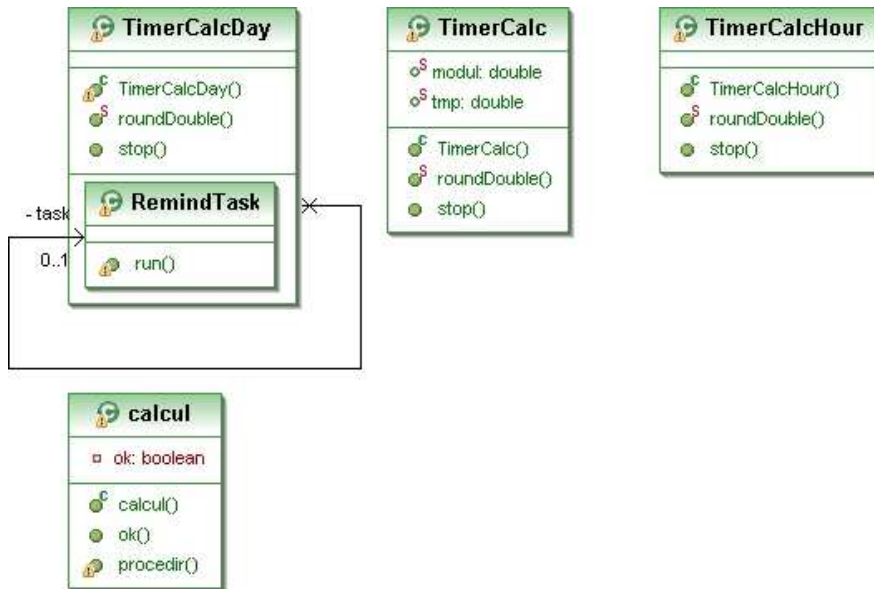


Figura 25. Diagrama de classes de calcSystem

caseSystem: ens permet solucionar casos.

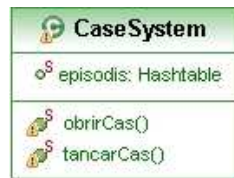


Figura 26. Diagrama de classes de caseSystem

listener: per a les connexions necessaries.



Figura 27. Diagrama de classes de listener

logicSystem: conté totes les classes del control.

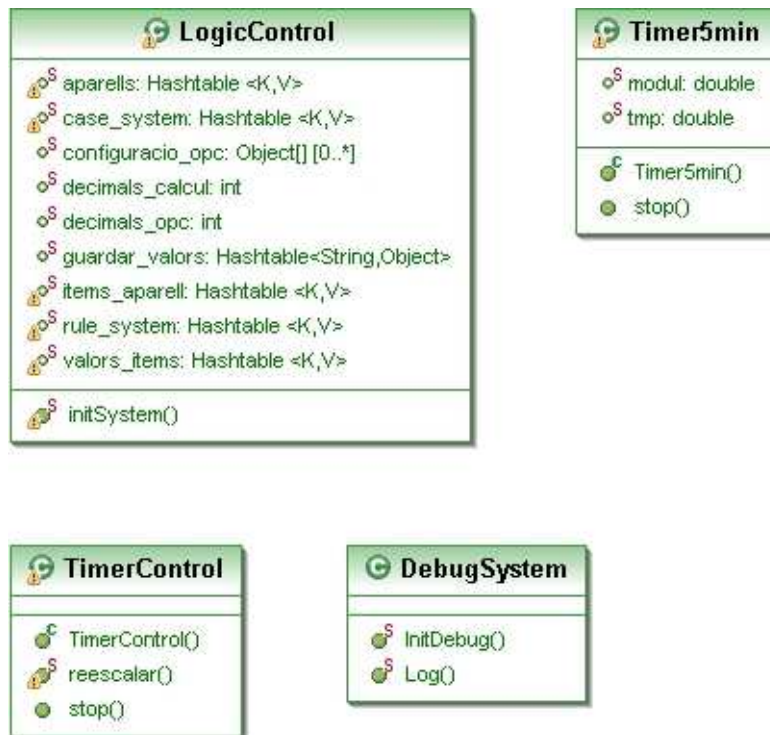


Figura 28. Diagrama de classes de logicSystem

mailSystem: conté les classes necessàries per a l'enviament de correus electrònics.



Figura 29. Diagrama de classes mailSystem

opcSystem: conté les classes que ens permeten la connexió amb el servidor OPC.

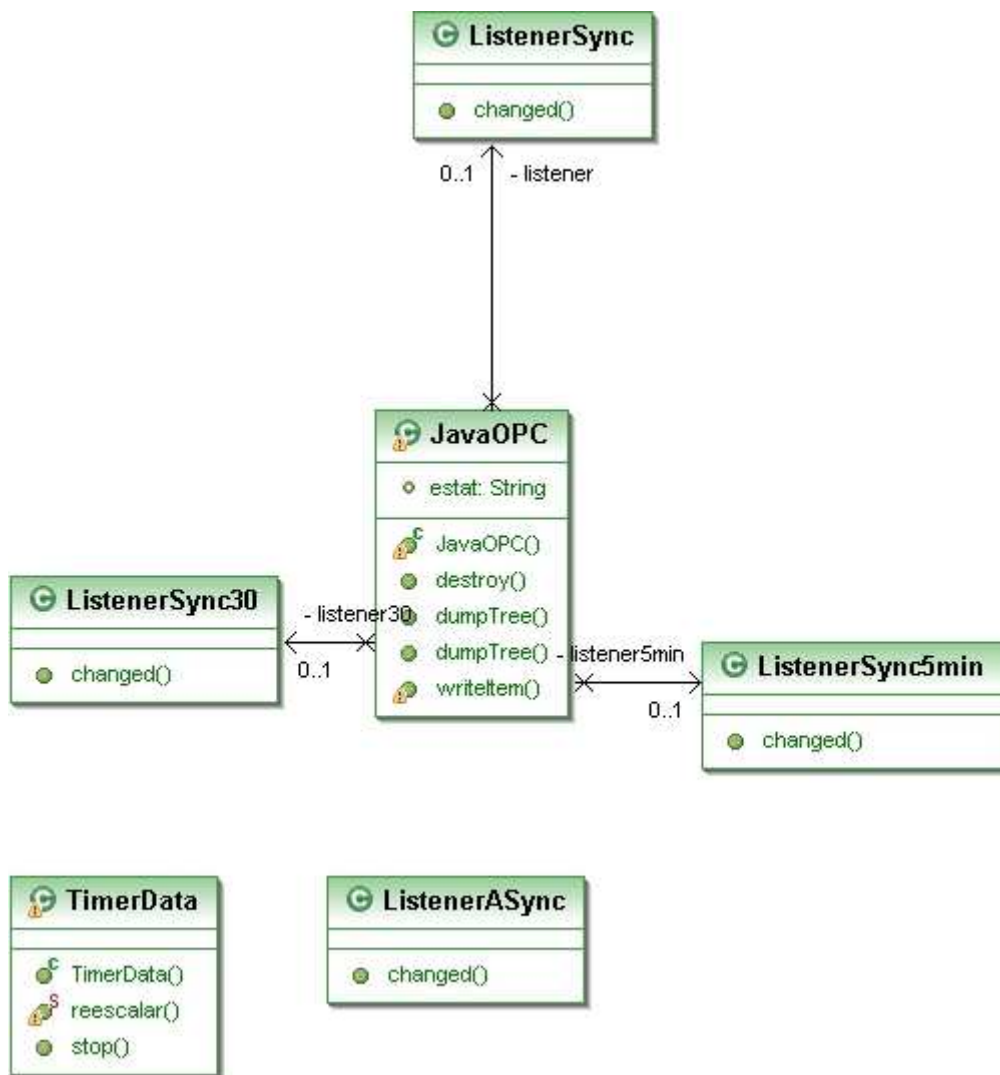


Figura 30. Diagrama de classes de opcSystem

persistenceSystem: conté les classes que ens permeten la connexió amb la base de dades.

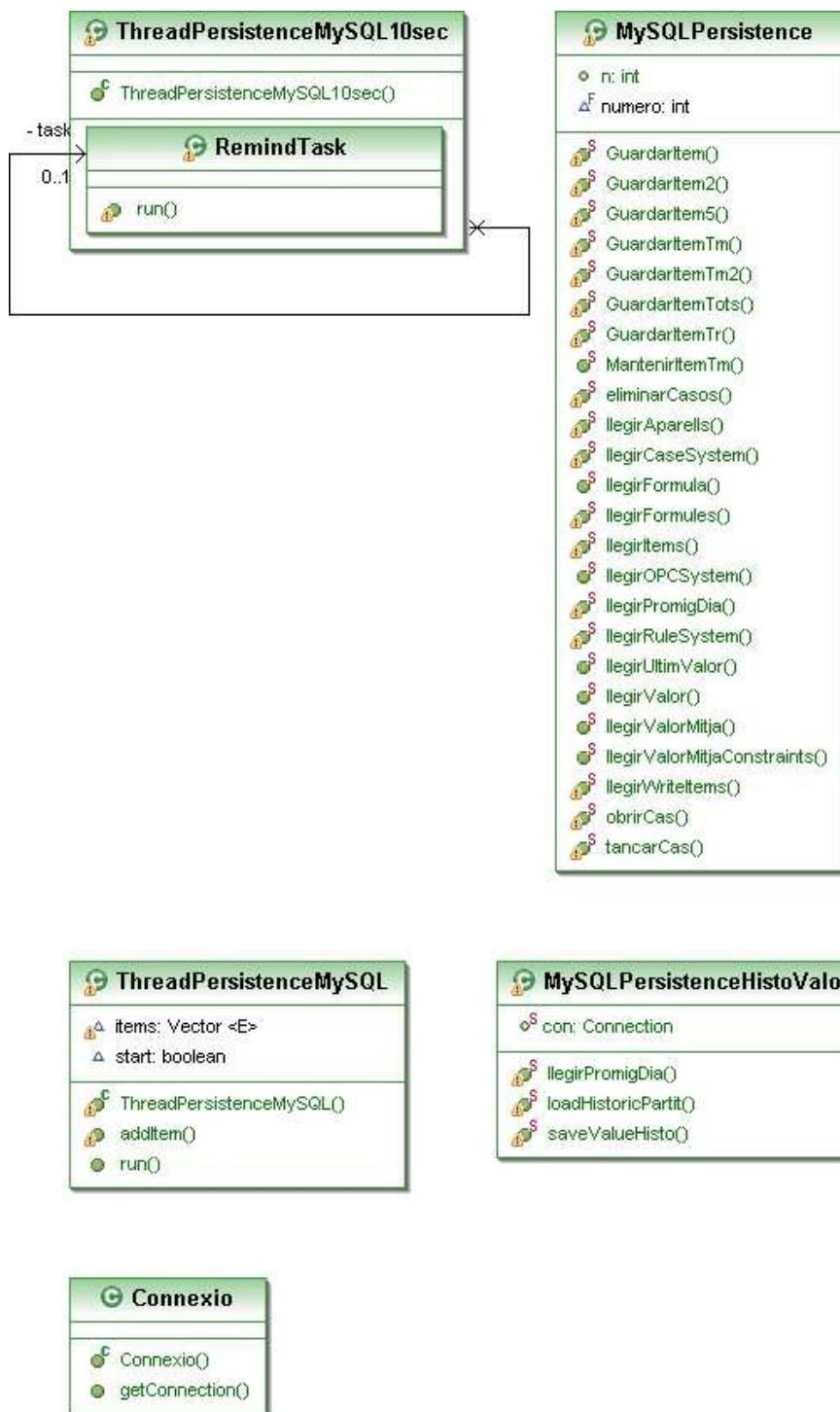


Figura 31. Diagrama de classes de persistenceSystem

ruleSystem: ens permetrà inicialitzar el que necessitem.



Figura 32. Diagrama de classes de ruleSystem

ruleSystem.loader: classes per a l'execució de la regal de control.

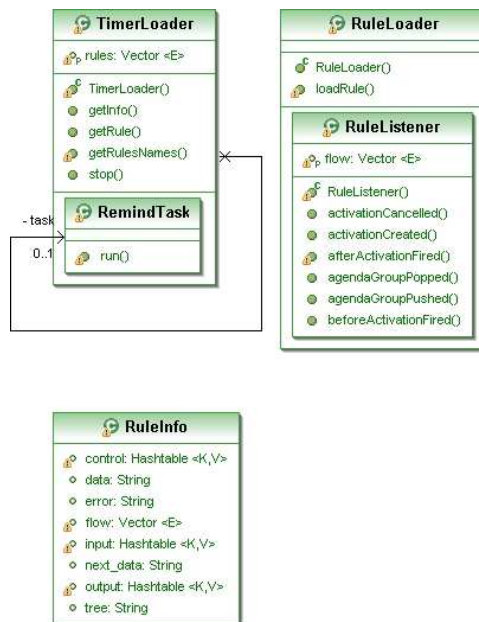


Figura 33. Diagrama de classes de ruleSystem.loader

ruleSystem.persistence: ens permetrà guardar informació.



Figura 34. Diagrama de classes de ruleSystem.persistence

ruleSystem.tree: ens permetrà interpretar la regla de control.

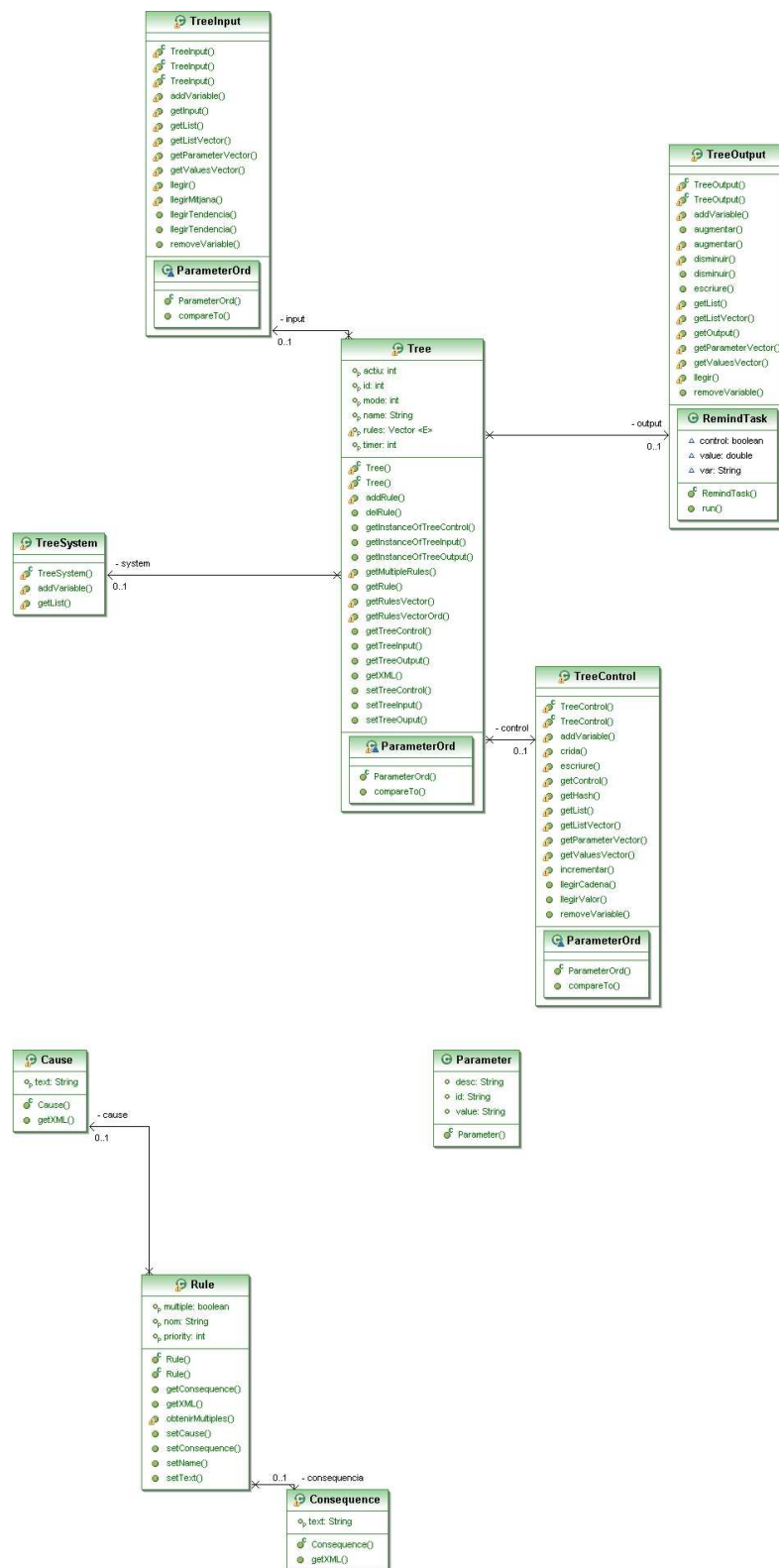


Figura 35. Diagrama de classes de ruleSystem.tree

ruleSystem.xml: ens permetrà estructurar el fitxer XML.



Figura 36. Diagrama de classes de ruleSystem.xml

utils: conté les classes utilitàries.

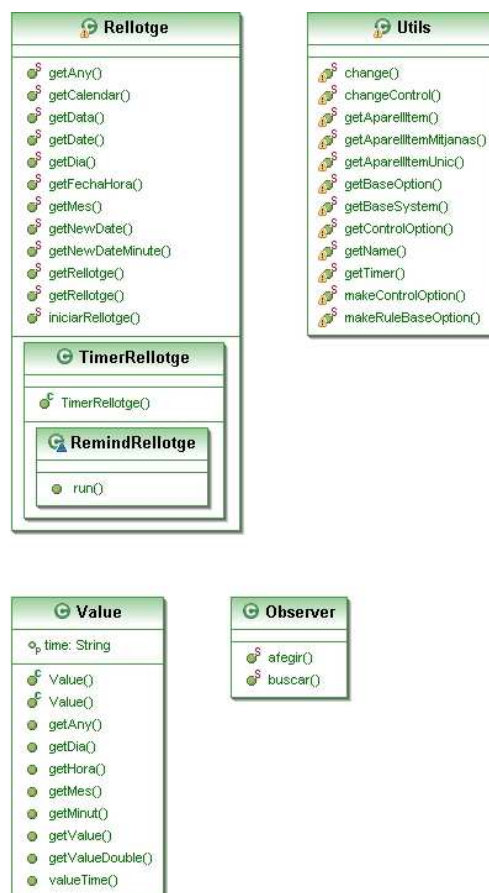


Figura 37. Diagrama de classes de utils

8.2. Aplicació web

En aquest apartat es detallarà les modificacions a realitzar a l'aplicació web des del punt de partida explicat a l'apartat 5.2.2. *Aplicació web*, dins de *Marc de treball i Conceptes previs*.

8.2.1. Anàlisi

L'aplicació web es comunica amb la base de dades per tal de mostrar les dades que obtenim del PLC, també els càlculs que es realitzen i els informes que podem descarregar.

Per a l'enviament de correus electrònics per informar de les alarmes actives, no és necessari mostrar res a través de l'aplicació web, ja que ja disposa d'un panell a l'inici de l'aplicació on podem visualitzar les alarmes actives en aquell moment (Figura 11).

Per als càlculs diaris, caldrà mostrar-los a l'aplicació web, mitjançant els seus valors guardats a la base de dades. Com es detalla a l'esquema següent:

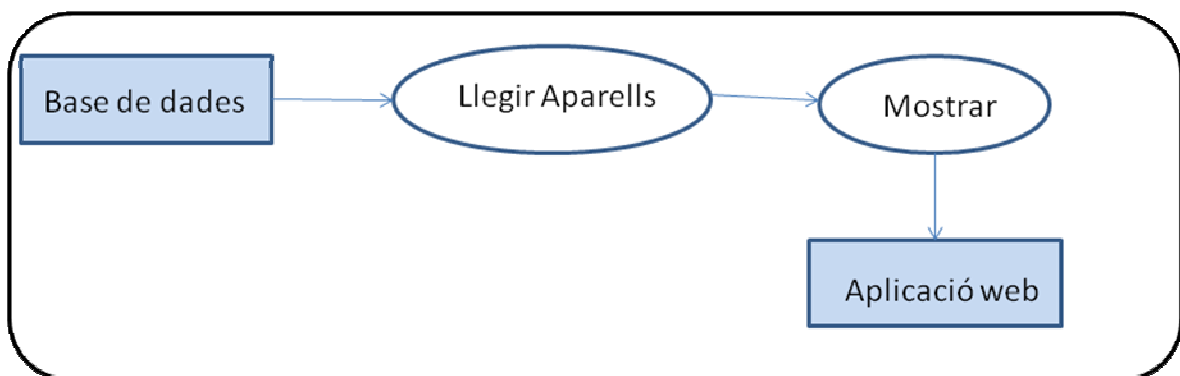


Figura 38. Model de processos dels càlculs diaris

Per a mostrar les dades de les regles de control haurem d'accedir a la base de dades que és on hi haurà tota la informació relacionada i els resultats obtinguts de les execucions, tal i com podem veure a l'esquema següent:

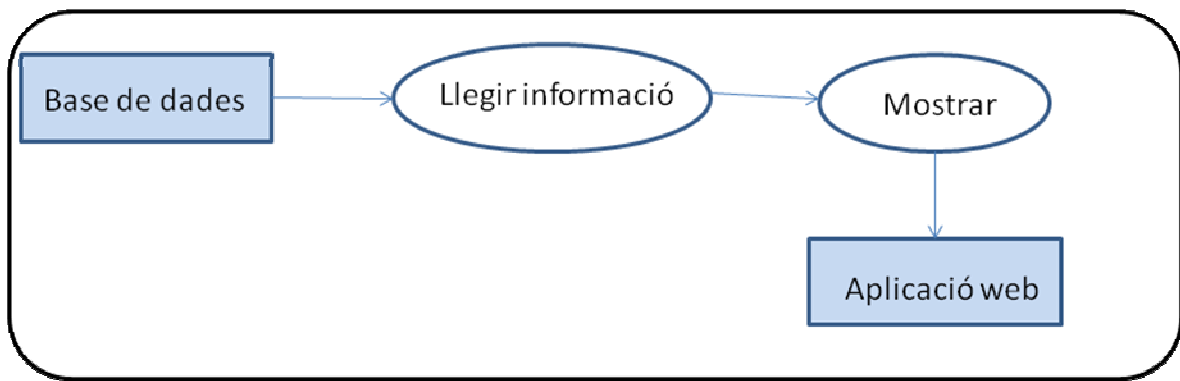


Figura 39. Model de processos de regles de control

8.2.2. Disseny

Per als càlculs diaris, un cop sabem els càlculs diaris que necessitem, es creen nous aparells a la pestanya *Units* a l'apartat *Calculations*. Es crea un aparell per cada càlcul diari que necessitem i que es mostrarà el valor que es calcularà dins l'aplicació "colmatar". Per tal de saber el valor corresponent de cada aparell l'aplicació web fa consultes a la base de dades, igual com si es tractés de qualsevol altre aparell que dona els valors instantanis, a diferència que d'aquests aparells només en tindrem un valor per cada dia. Es pot veure a la imatge següent com hi ha càlculs diaris:

Unitats » General	
Veure: Tots Sensors Equipment Data Lab Set Points Alarms Calculations	
Litres permeat diari (últim valor) (més informació)	4 L/dia
Litres rentat diari (últim valor) (més informació)	0 L/dia
MLSS diaris (més informació)	22.3173 g SST/L
OD aerobí diari (més informació)	1.0934 mg O2/L
Pendent permeabilitat instantània 14 dies (més informació)	0 null
Pendent permeabilitat instantània 4 dies (més informació)	-53.1465 null
Pendent TMP 4 dies (més informació)	-0.7705 null
Pendent TMP diari (més informació)	0 null
Permeabilitat Corregida (més informació)	621.37385 L/(m2*h*bar)
Permeabilitat corregida diària (més informació)	489.7396 L/(m2*h*bar)
Permeabilitat Instantània (més informació)	500 L/(m2*h*bar)
Permeabilitat instantània diària (més informació)	541.6068 L/(m2*h*bar)
Recovery (més informació)	0 %
Rejection (més informació)	69.89619 %
SADm (més informació)	1.75 m/h

Figura 40. Càlculs diaris

De les regles de control a través de la nova pestanya *Rules* podem visualitzar la informació necessària, document XML, imatges de l'arbre de decisions, conseqüències i resultats de l'última execució. Tot extret de la base de dades en les taules corresponents.

En la imatge següent podem veure la pantalla de la pestanya Rules:

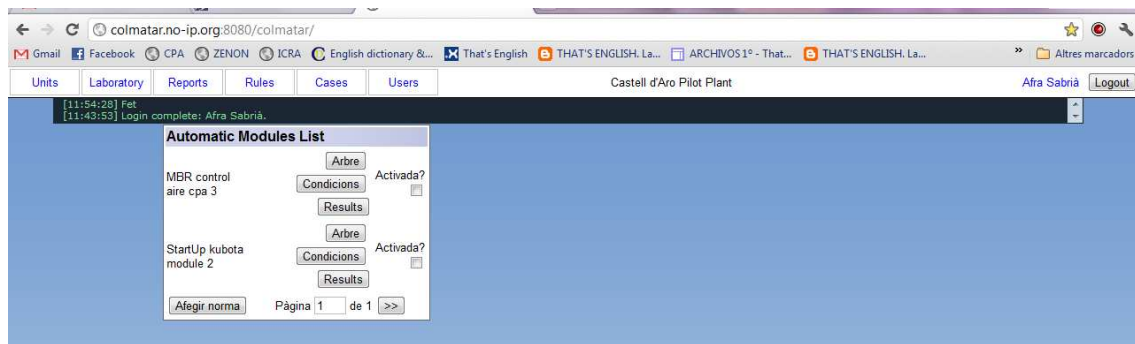


Figura 41. Visualització pestanya Rules

Si ens posicionem sobre del nom de la regla de control, en aquest cas, *MBR control aire cpa 3* i cliquem, podem visualitzar la id de la regla de control, el nom i el fitxer xml, com es veu a continuació:

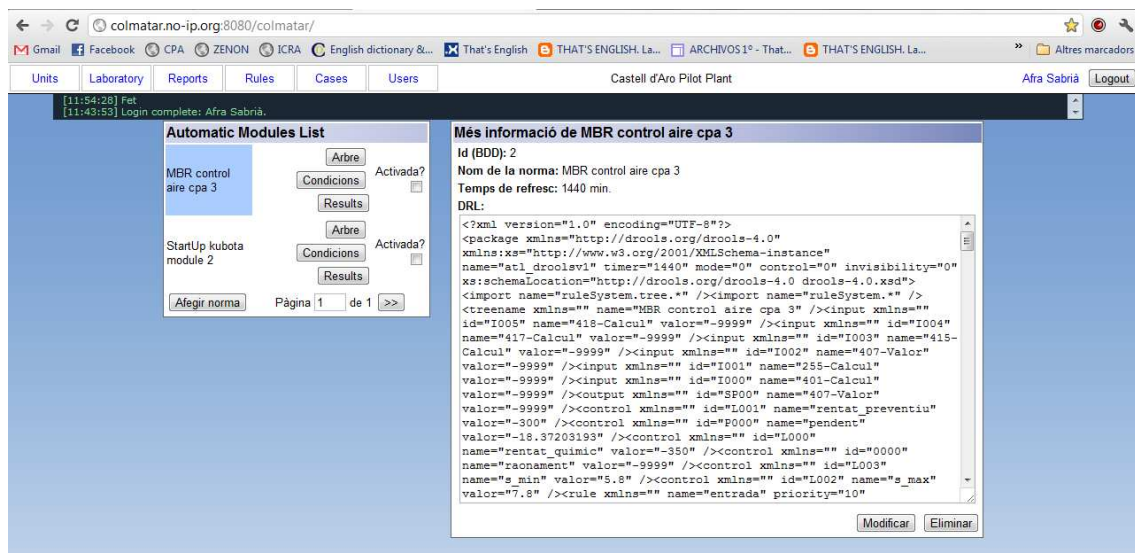


Figura 42. Visualització del XML de la regla de control

Si cliquem sobre el botó *Arbre* podem visualitzar l'arbre de decisions corresponent a aquella regla de control, com es veu a la imatge següent:

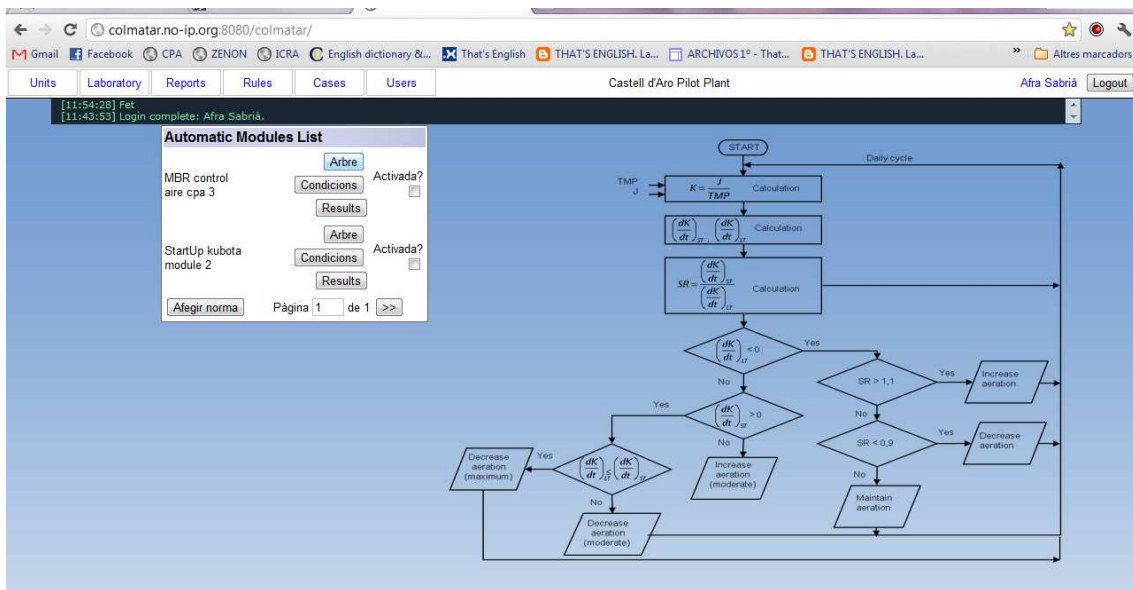


Figura 43. Visualització de l'Arbre

Si cliquem sobre el botó *Condicions* podem visualitzar les condicions i les seves conseqüències que s'esperen de l'execució de la regla de control, com es veu a la imatge següent:

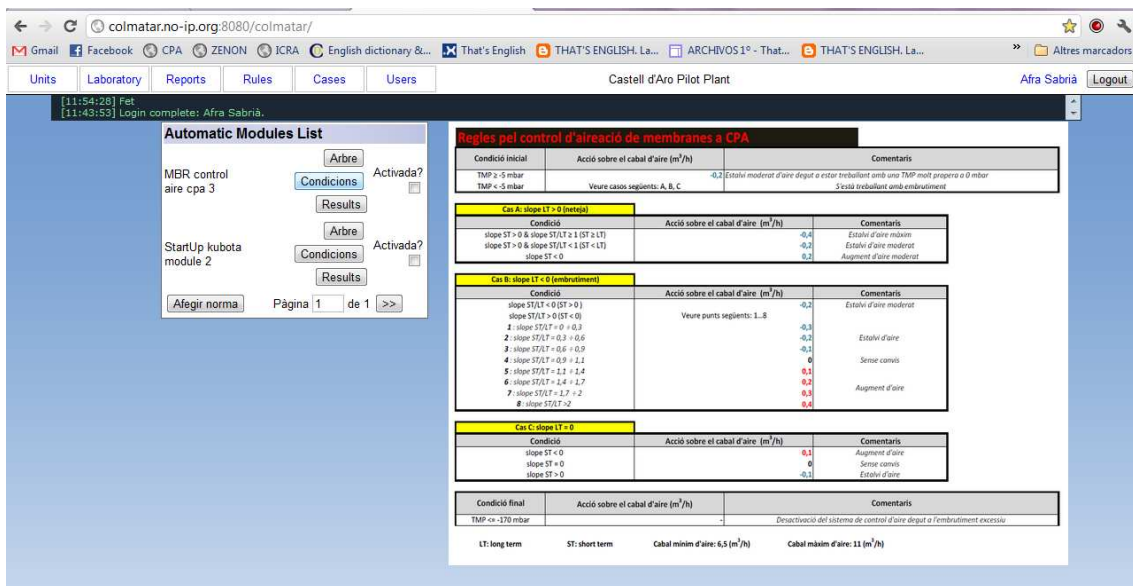


Figura 44. Visualització de les Condicions i Conseqüències

Com podem veure a la imatge que hi ha a continuació, si premem el botó *Results* podem visualitzar els resultats obtinguts de l'última execució de la regla de control.

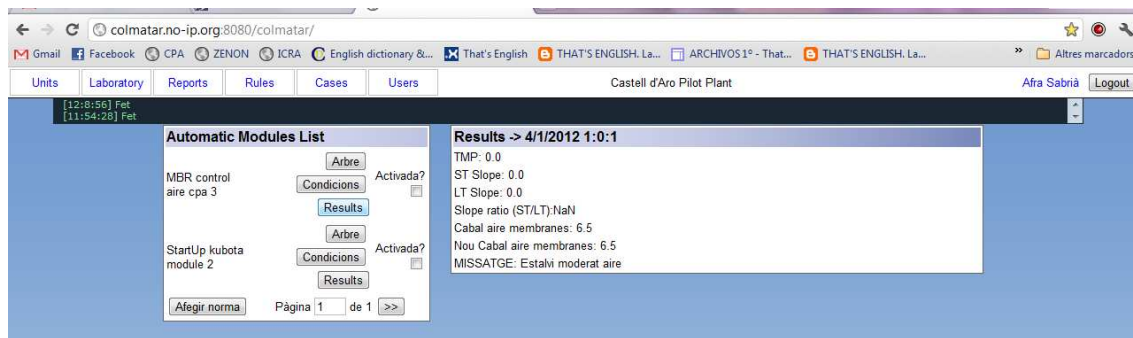


Figura 45. Visualització de Results (última execució)

A l'aplicació web caldrà afegir per a poder visualitzar la pestanya *Rules* amb totes les seves opcions:

Fitxer js:

- rules

Fitxer jsp:

- servidor_normes

I modificar perquè es vegi en el panell inicial:

Fitxers js:

- panell
- interfície

Fitxer jsp:

- servidor_general

Fitxer css:

- index

Fitxer html:

- index

8.3. Base de dades

En el mòdul d'enviar correus electrònics no ens caldrà crear ni modificar res de la base de dades ja que es fa tot directament des de l'aplicació *colmatar*.

Per als càlculs diaris, no s'haurà de crear ni modificar res ja que es crearan nous aparells de *Càlcul* amb una entrada a la base de dades diària, que guardarà el valor calculat. Es guardarà a la mateixa taula (*histovalor_2012_01* per exemple) on es guarden la resta dels aparells i es podrà buscar el seu valor tal i com es fa amb la resta dels aparells a través del seu identificador.

Per a les regles de control s'hauran de crear algunes taules:

- rulebase: on es guardaran totes les regles de control de què disposem en format XML amb un identificador, el seu nom i si està activa o no.
- rulebaseinfo: es guardarà la data i la hora de l'execució d'una regla de control i el seu nom.
- ruleinfovalues: es guardarà tota la informació relacionada amb l'execució de les regles de control, el nom, la data i hora d'execució, el tipus de paràmetre (input, control, flow, output, etc.), la variable utilitzada i el seu valor.

8.4. Disseny de les regles de control

Les regles de control ens venen definides per uns objectius que es volen assolir i experimentar. Són dissenyades en la part teòrica pels investigadors i els membres del Grup de MBR i la part pràctica o de disseny informàtic que me n'encarrego jo. Es dissenyen tenint en compte del que disposem, el que cal modificar o crear per tal de complir els objectius principals.

Per a posar en funcionament el nou mòdul de control automàtic, s'han creat dues regles de control, una és la *MBR Control Aire* i l'altra *Startup Kubota Module*, que han donat suport a les dues tesis anteriorment esmentades a l'apartat 5.3.

Membres del grup juntament amb els investigadors principals creen un arbre de decisions a partir del qual es dissenyarà una nova regla de control i que es duran a terme unes proves a la planta pilot de Castell d'Aro (CPA) entre d'altres.

8.4.1. MBR Control Aire

L'objectiu principal d'aquesta regla de control és el d'automatitzar el control de l'aire per tal de minimitzar-lo i per tant poder obtenir un estalvi en el consum d'energia.

8.4.1.1. Arbre

Partint d'un arbre de decisions com el que es mostra a continuació i que és l'esquema de les decisions a prendre, es desenvolupa coneixent també altres dades, la regla de control.

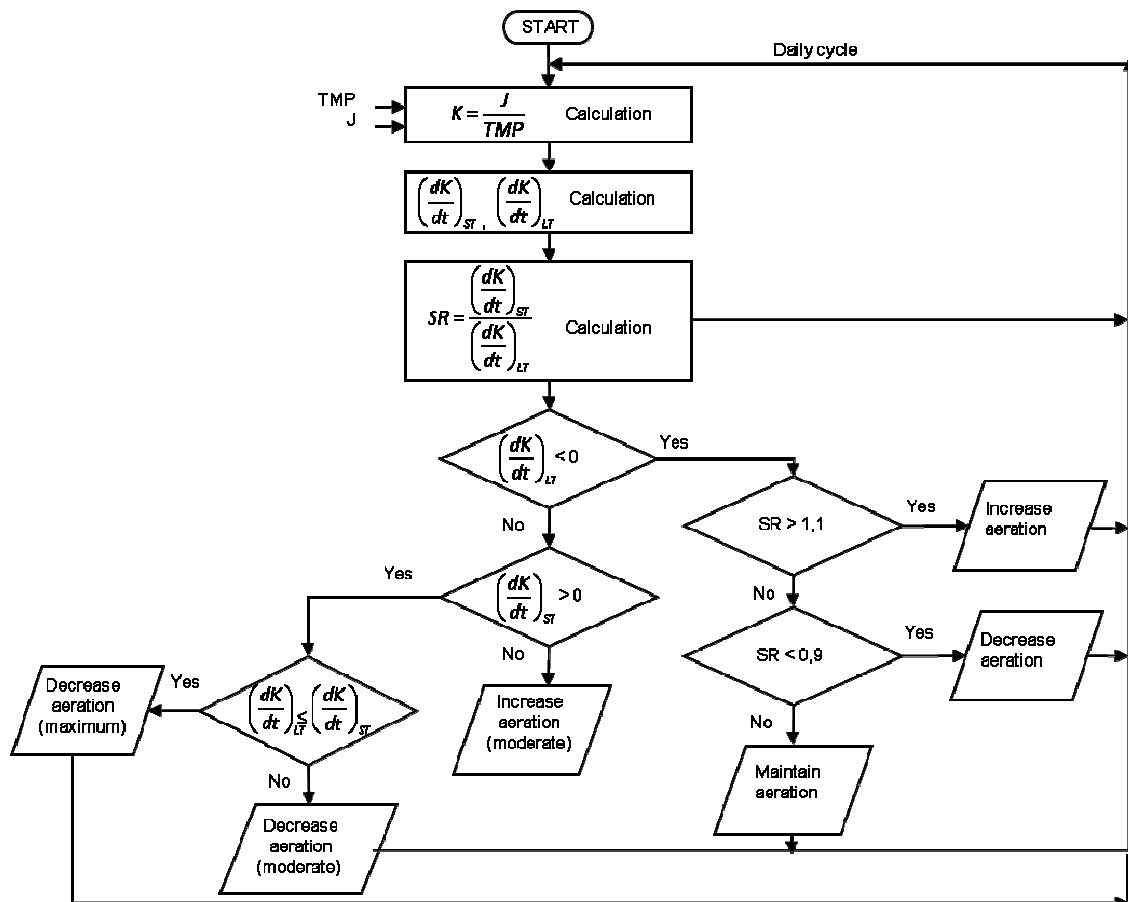


Figura 46. Arbre de decisions MBR Control Aire CPA

Aquest arbre de decisions es completa amb les corresponents dades de configuració, que seran diferents a cada planta segons les seves característiques, en aquest cas per la planta pilot de CPA:

Regles pel control d'aireació de membranes a CPA		
Condicció inicial	Acció sobre el cabal d'aire (m ³ /h)	Comentaris
TMP ≥ -5 mbar TMP < -5 mbar	-0,2 Veure casos següents: A, B, C	Estalvi moderat d'aire degut a estar treballant amb una TMP molt propera a 0 mbar S'està treballant amb embrutiment
Cas A: slope LT > 0 (neteja)		
Condicció	Acció sobre el cabal d'aire (m ³ /h)	Comentaris
slope ST > 0 & slope ST/LT ≥ 1 (ST ≥ LT)	-0,4	Estalvi d'aire màxim
slope ST > 0 & slope ST/LT < 1 (ST < LT)	-0,2	Estalvi d'aire moderat
slope ST < 0	0,2	Augment d'aire moderat
Cas B: slope LT < 0 (embrutiment)		
Condicció	Acció sobre el cabal d'aire (m ³ /h)	Comentaris
slope ST/LT < 0 (ST > 0)	-0,2	Estalvi d'aire moderat
slope ST/LT > 0 (ST < 0)		
1 : slope ST/LT = 0 ÷ 0,3	-0,3	
2 : slope ST/LT = 0,3 ÷ 0,6	-0,2	Estalvi d'aire
3 : slope ST/LT = 0,6 ÷ 0,9	-0,1	
4 : slope ST/LT = 0,9 ÷ 1,1	0	Sense canvis
5 : slope ST/LT = 1,1 ÷ 1,4	0,1	
6 : slope ST/LT = 1,4 ÷ 1,7	0,2	Augment d'aire
7 : slope ST/LT = 1,7 ÷ 2	0,3	
8 : slope ST/LT > 2	0,4	
Cas C: slope LT = 0		
Condicció	Acció sobre el cabal d'aire (m ³ /h)	Comentaris
slope ST < 0	0,1	Augment d'aire
slope ST = 0	0	Sense canvis
slope ST > 0	-0,1	Estalvi d'aire
Condicció final	Acció sobre el cabal d'aire (m ³ /h)	Comentaris
TMP ≤ -170 mbar	-	Desactivació del sistema de control d'aire degut a l'embrutiment excessiu

LT: long term ST: short term Cabal mínim d'aire: 6,5 (m³/h) Cabal màxim d'aire: 11 (m³/h)

Figura 47. Dades pel control

8.4.1.2. De què disposem i què necessitem?

Disposem dels valors dels aparells que ens venen donats pel PLC, però caldrà estudiar quins valors necessitem per a poder dissenyar la regla de control i quins càlculs seran necessaris de realitzar.

Per tal de poder analitzar millor quins valors són necessaris serà necessari reunir-se amb els responsables del disseny de l'arbre de decisions per tal de saber quines dades necessiten visualitzar i quin és l'objectiu del disseny de la regla de control.

Així doncs, tal i com es veu a la Figura 47, veurem que necessitem una *slope ST* (short term) que serà una pendent a curt termini i que es decideix que sigui la pendent de la *Permeabilitat Instantània* (valor que ens ve donat pel PLC) i s'acorda que sigui un pendent de 4 dies. Per a realitzar el càlcul d'aquest pendent, obtindrem una mitjana diària dels valors donats a cada cicle de permeat vàlid (serà vàlid si dura més de 2 minuts), d'aquesta manera farem el càlcul de l'*slope ST*. També necessitarem un *slope LT* (long term) que serà una pendent a llarg termini de la *Permeabilitat Instantània* i que es decideix que sigui un pendent de 14 dies calculat a partir de la mitjana diària de la PMI (permeabilitat instantània) dels cicles de cada dia, igual com *slope ST*. Caldrà saber també una mitjana diària dels valors de TMP (pressió

transmembrana) dels cicles vàlids que hi ha durant el dia. A part també es voldrà visualitzar el valor de PMI del dia.

Tots aquests càlculs que es realitzaran cada dia de mitjanes i pendents, estaran implementats dins a la nova classe de l'aplicació *colmatar*, *TimerCalcDay* que utilitza *MySQLPersistenceHistoValor* per tal de poder detectar els cicles vàlids que hi ha durant el dia i d'aquesta manera obtenir-ne les mitjanes pertinents.

L'aparell del qual haurem de modificar el seu valor per tal d'augmentar o disminuir l'aire podem accedir-hi a través del seu identificador i la regla de control ha de poder canviar-ne el seu valor.

8.4.2. StartUp Kubota Module

L'objectiu principal d'aquesta regla de control és controlar i obtenir els nivells òptims de concentració de la biomassa aconsellats pel fabricant, des de la posada en funcionament de la planta pilot i alhora també accelerar l'eliminació biològica de nutrients i obtenir amb el mínim de temps possible el flux o cabal de filtració desitjat de treball.

8.4.2.1. Arbre

Donat l'arbre de decisions que es mostra a continuació, on podem veure que Flux, MLSS, TMP, FR-TMP i Filterability són dades necessàries per a la interpretació de l'arbre, caldrà identificar si aquestes vénen donades per un aparell que pertany a la planta pilot o bé s'han d'obtenir a partir d'algun càlcul.

L'arbre ens mostra les comprovacions que caldrà fer dels valors d'aquests ítems i segons aquests valors les decisions que caldrà prendre.

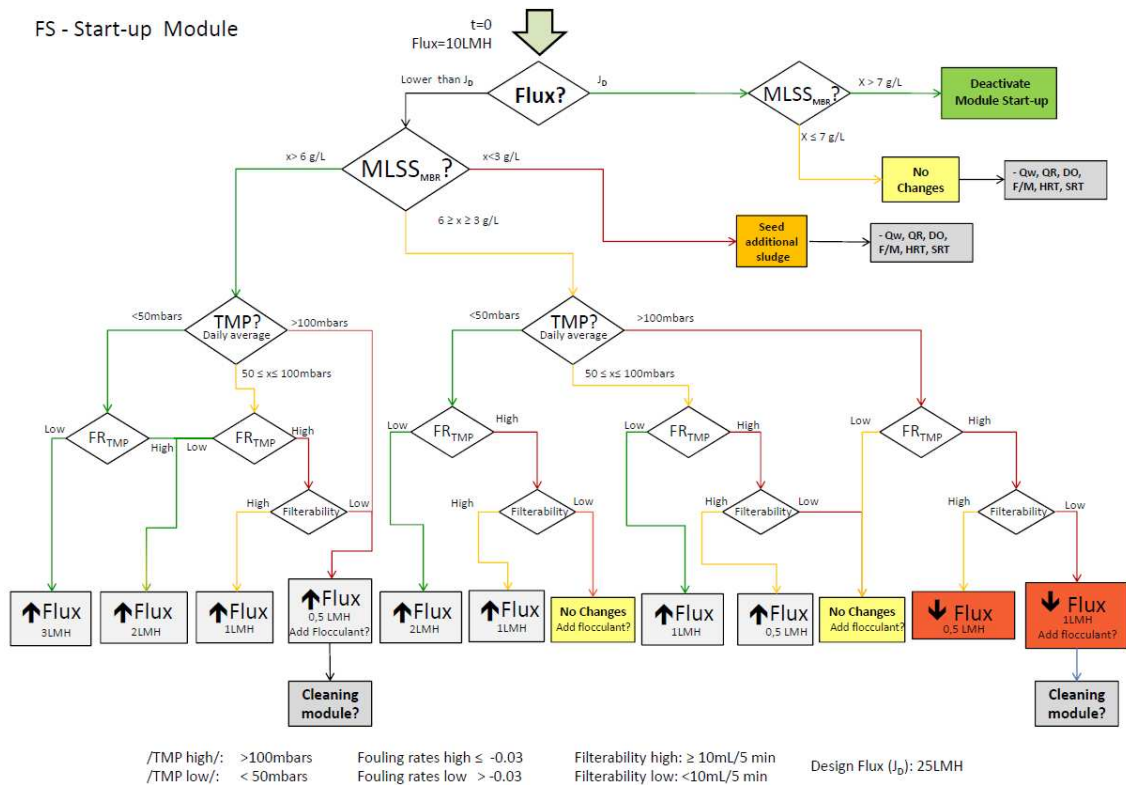


Figura 48. Arbre de decisions StartUp Kubota Module

8.4.2.2. De què disposem i què necessitem?

Disposem d'aparells que ens donen el seu valor a través del PLC. Per tal de conèixer amb més profunditat i les necessitats d'aquest arbre de decisions caldrà reunir-se amb els responsables del disseny de l'arbre.

El *Flux* que necessitem ja ens ve donat el seu valor a partir d'uns càlculs instantanis obtinguts a partir de valors directament des del PLC (cada 10 segons). Necessitarem la mitjana diària de TMP per tant caldrà realitzar un nou càlcul diari. També ens caldrà FR-TMP (*fouling rates de TMP*) i que haurem de calcular diàriament, amb els valors de TMP durant un dia. Obtinguts a partir de tots els cicles vàlids, fer el càlcul del pendent durant aquell dia. Ens caldrà saber també la mitjana diària de la *Filterability* i també la dels *MLSS (Mixed Liquor Suspended Solids)*.

Per tant els càlculs diaris que necessitem s'hauran de calcular dins la classe *TimerCalcDay* utilitzant *MySQLPersistenHistoValor* per tal de detectar de cada aparell que ens dona el valor directe del PLC, els cicles de permeat que hi ha durant aquell dia i obtenir-ne les mitjanes diàries.

L'arbre de decisions ens mostra que caldrà augmentar o disminuir el *Flux*. Com que el Flux s'obté a partir d'un càlcul instantani (cada 10 segons igual com la resta de dades), per tal de modificar-lo haurem de canviar el valor d'un altre aparell, la *Consigna de cabal de treball*.

9. Implementació i proves

En aquest apartat s'explica com s'ha realitzat la implementació i proves de les regles de control dissenyades.

9.1. MBR Control Aire

Afegir a la classe *MySQLPersistenceHistoValor*, els càlculs diaris que necessitem.

Segons s'acorda a les reunions amb els membres del grup, el càlcul diari a fer serà la mitjana diària de totes les mitjanes de cada cicle que hi hagi durant un dia.

Un cicle es considera uns minuts determinats en què la planta està permeant, seguidament fa un temps de relaxació, així doncs considerem un cicle els minuts en què està permeant, i serà vàlid si aquest dura més de dos minuts.

Per al càlcul diari es fa una detecció del cicle mitjançant la *Bomba de permeat* (aparell que ens permet saber que quan la seva senyal és 1 està permeant i quan és 0 està relaxant), aparell que tenim identificat al nostre sistema amb id 197 i amb una adreça de memòria determinada.

Un cop podem detectar el cicle se n'agafen totes les dades i es filtren segons un criteri establert per eliminar valors que considerariem no correctes, després es calcula una mitjana de cada cicle. A partir de la mitjana de cada cicle i sabent quants cicles hi ha durant un dia sencer, podem saber la mitjana diària.

Aquest tipus de càlcul l'utilitzem per a conèixer la *TMP diària* que identifiquem al nostre sistema amb la id 255 i també per conèixer la *PMI diària* que identifiquem amb id 415.

Per tal de calcular els pendents *slope ST* i *slope LT*, on considerem ST 4 dies i LT 14 dies, farem una regressió de 4 dies comptant l'actual i anant endarrere, i igualment amb els 14 dies, a partir de les mitjanes diàries de *PMI*.

Un cop tenim totes les dades precalculades que necessitem, havent comprovat que cada dia es calculen aquests paràmetres i un cop verificat que el càlcul és correcte, es pot procedir a dissenyar la regla de control.

9.1.1. Disseny de la regla de control

Per al disseny de la regla de control utilitzem una aplicació ja existent i mencionada anteriorment.

Primerament hem d'identificar quins paràmetres necessitem d'entrada, quins paràmetres de control i quins de sortida. Per tant tindrem que necessitem d'entrada:

- slope ST: aparell id 417.
- slope LT: aparell id 418.
- TMP diària: aparell id 255.
- PMI diària: aparell id 415.
- Consigna cabal aire membranes: aparell id 407.

Els paràmetres d'entrada els introduïrem com es veu a la imatge següent:

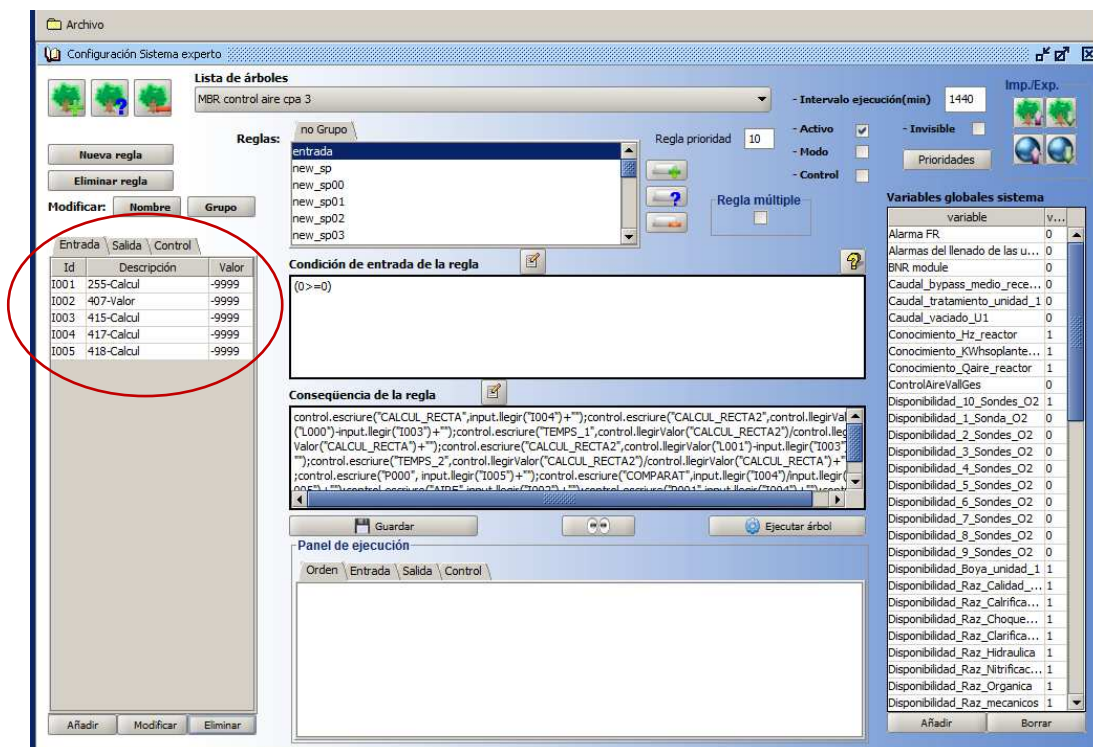


Figura 49. Paràmetres d'entrada

Com a paràmetres de control necessitarem:

- Cabal mínim que es pot assolir
- Cabal màxim que es pot assolir

Totes les dades que posem al control seran perquè sigui més fàcil modificar el seu valor sense haver de modificar el codi. En aquest cas tindrem un mínim i un màxim de *Cabal d'aire de membranes* que es podrà assolir, però podria ser que degut a les proves reals que es duguin a terme aquest mínim i màxim s'hagin de modificar. A continuació es mostra com s'introdueixen els paràmetres de control:

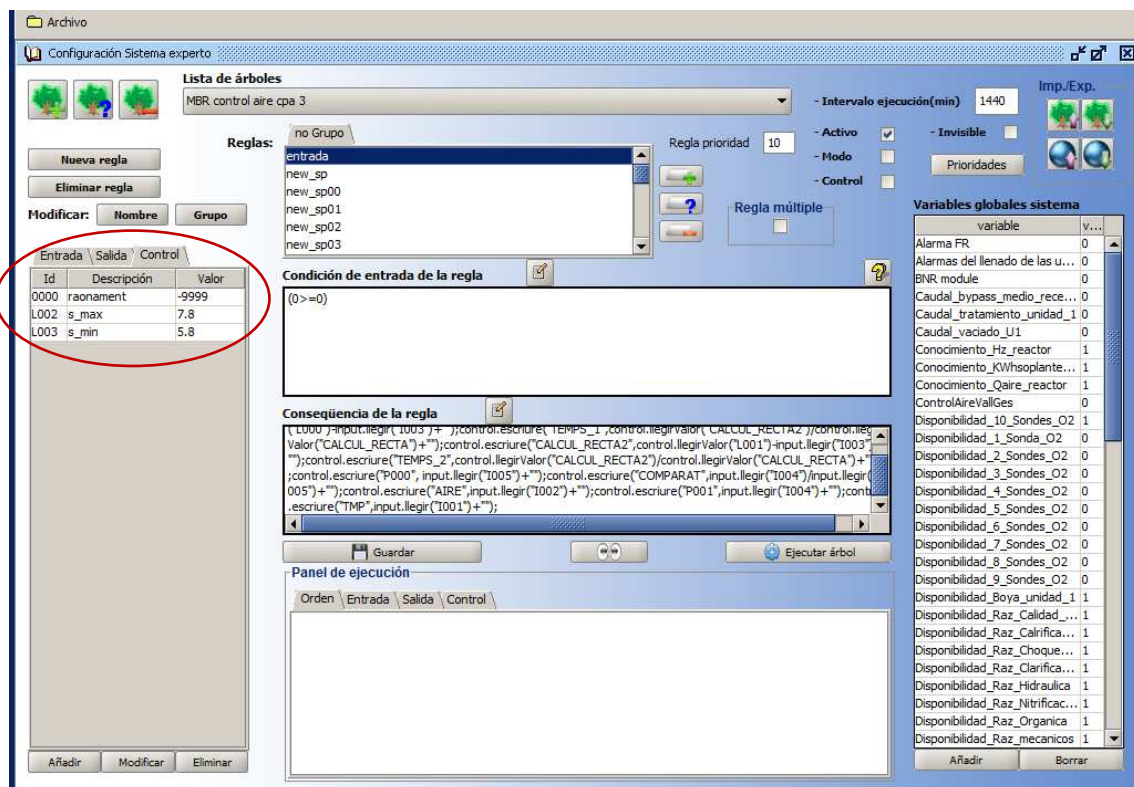


Figura 50. Paràmetres de Control

Paràmetres de sortida:

- Consigna cabal d'aire membranes: aparell id 407

Només ens caldrà modificar el valor d'un aparell, en aquest cas la Consigna de cabal d'aire de membranes, tot i que s'hauran de mostrar altres dades, però es mostrarà dins la mateixa implementació de la regla de control. A la imatge següent podem veure els paràmetres de sortida:

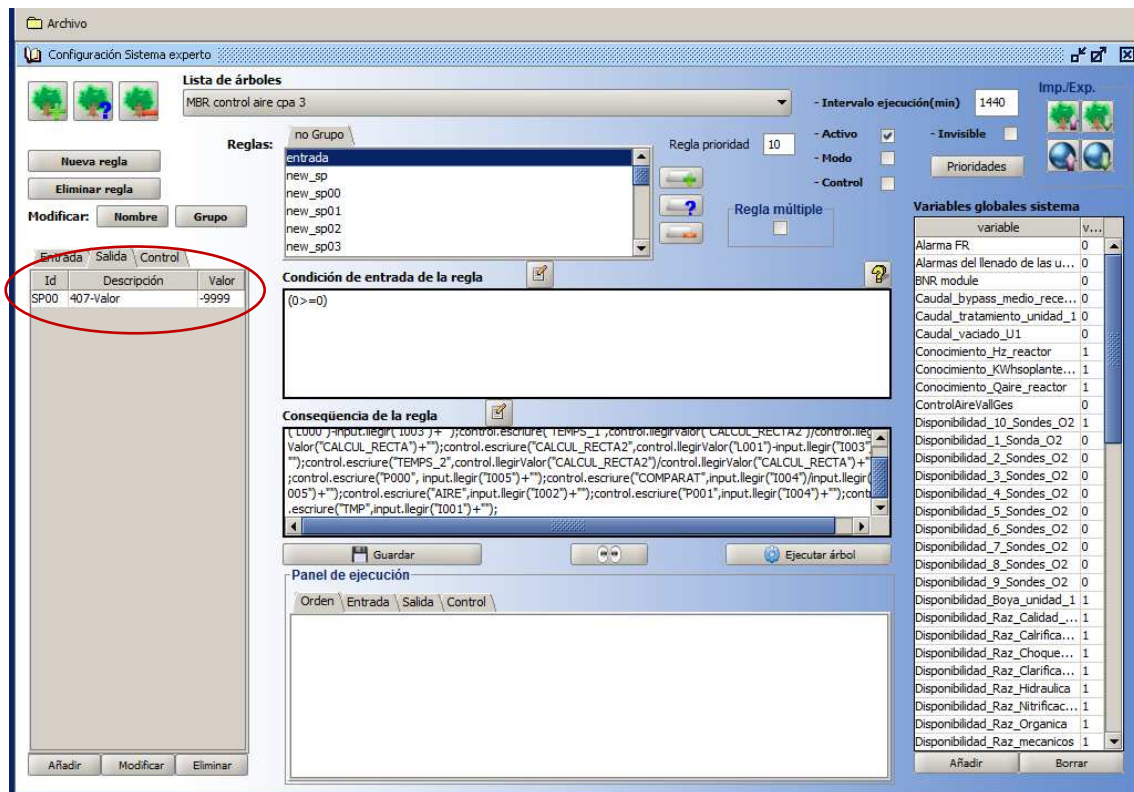


Figura 51. Paràmetres de Sortida

Per a la implementació bàsicament tindrem diferents parts a definir:

- entrada: on identificarem cada aparell de l'entrada que hem de conèixer el seu valor per un nom que ens permetrà treballar més fàcilment a la implementació:
- new_spXX: es definiran totes les branques possibles de la implementació de la regla de control amb les seves condicions d'entrada i la corresponent conseqüència.
- output: on controlarem que el valor o valors a modificar està dins els límits que li marquem.
- write_rule: on es realitza el canvi de valor de l'aparell que hem de modificar.
- sortida: es defineix tot el que volem que es mostri a l'aplicació web com a resultat de l'execució de la regla de control.

Cada part tindrà un valor per indicar-ne la prioritat de la seva execució.

Entrada:

- Condicció: $0 >= 0$ perquè volem que entri sempre.

- Conseqüència: llegirà els valors que li donem a Entrada.
- Prioritat: indica la prioritat d'execució.

A la següent imatge veiem l'Entrada:

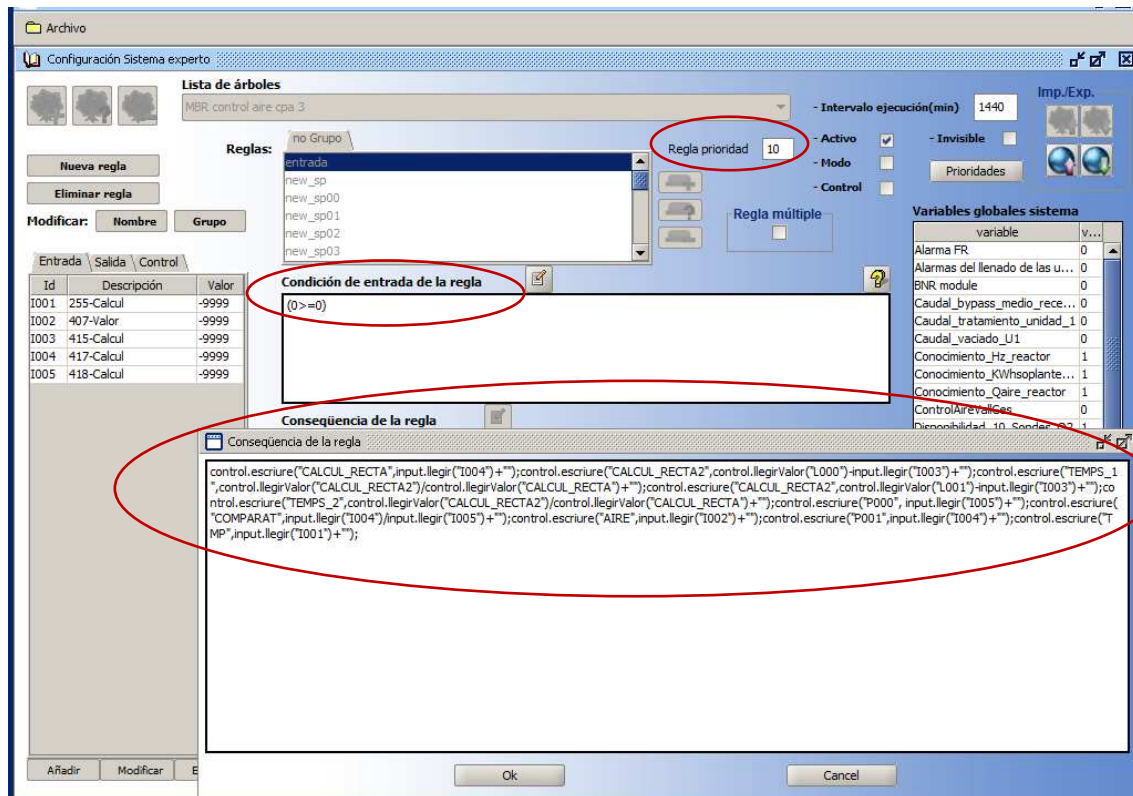


Figura 52. Condició i Conseqüència d'entrada

New_sp:

Exemple d'una de les branques de la regla de control.

- Condició: condicions d'entrada segons els paràmetres llegits.
- Conseqüència: conseqüència si es compleix la condició d'entrada.
- Prioritat: prioritat d'execució.

A la següent imatge veiem un exemple d'un dels new_sp:

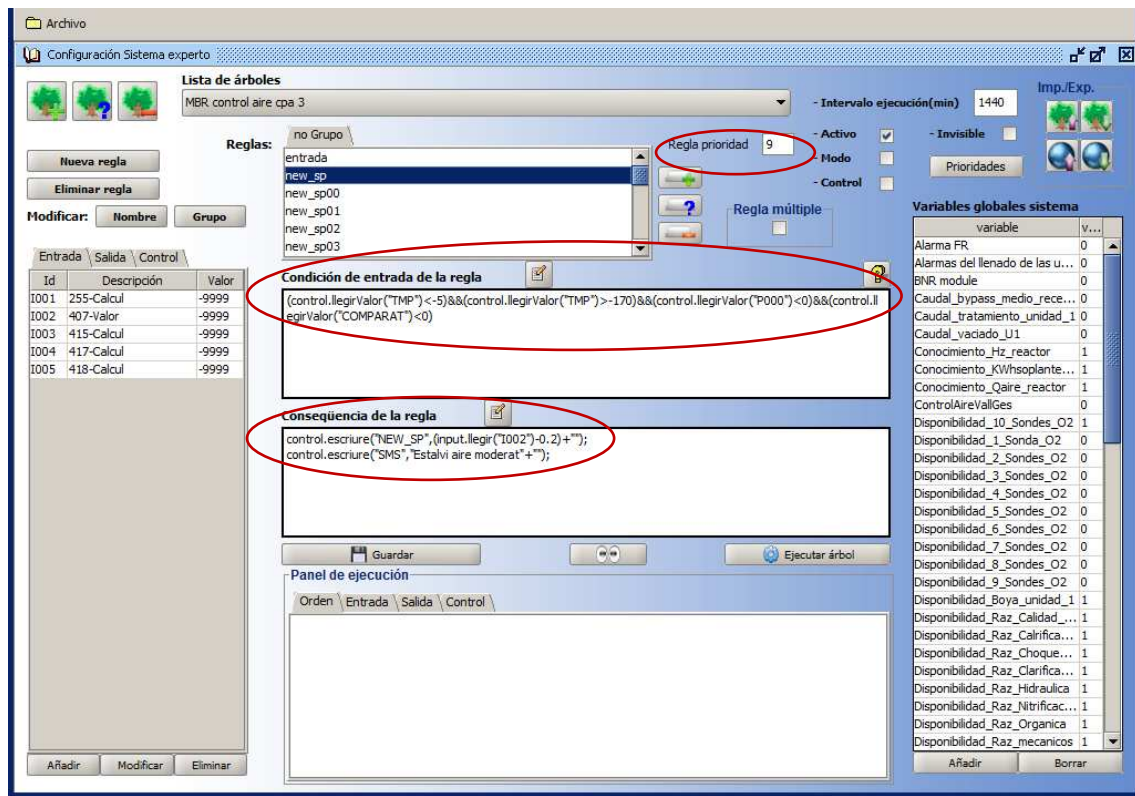


Figura 53. Una de les branques (new_sp) de la regla de control

A la Figura 53 on es mostra una de les branques de la regla de control, en concret new_sp, la qual correspondria a la comprovació que es mostra amb les imatges que hi ha a continuació:

Condició inicial	Acció sobre el cabal d'aire (m³/h)	Comentaris
TMP < -5 mbar	Veure casos següents: A, B, C	S'està treballant amb embrutiment.

Figura 54. Comprovació del valor de TMP

S'ha de complir que "COMPARAT" tal i com es veu a la Figura 53, que correspon a slope ST/slope LT, sigui més petit que 0. Per tant tindrem que slope LT<0 i que ST>0. Podem veure que l'actuació serà baixar 0.2 m³/h el cabal d'aire i mostrar el missatge "Estalvi d'aire Moderat", com podem veure a la imatge següent:

Cas B: slope LT < 0 (embrutiment)		
Condició	Acció sobre el cabal d'aire (m³/h)	Comentaris
slope ST/LT < 0 (ST > 0)	-0,2	Estalvi d'aire moderat

Figura 55. Accions a realitzar

Output:

- Condició: Condició per comprovar que el canvi que ha de realitzar no passa del màxim de cabal d'aire que es defineix, en aquest cas 11.

- Conseqüència: Conseqüència si es compleix la condició.
- Prioritat: prioritat d'execució.

La següent imatge mostra un dels output de la regla de control:

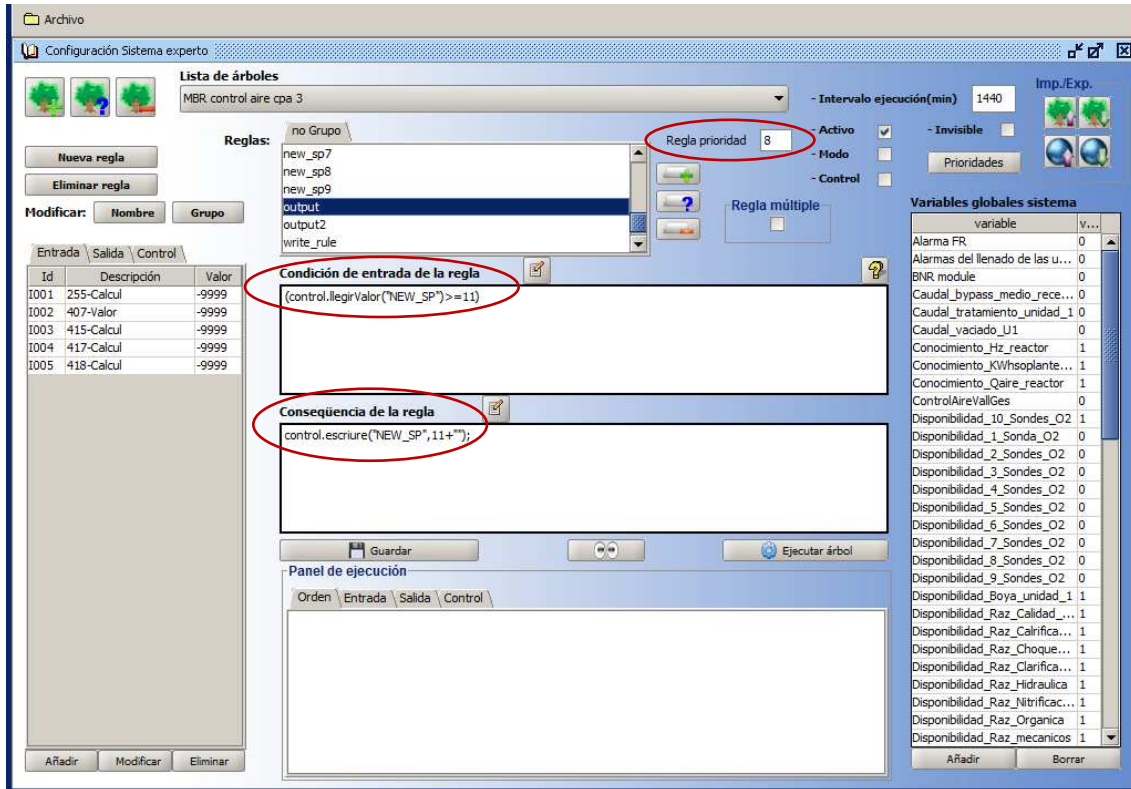


Figura 56. Exemple d'output

Write rule:

- Condició: el canvi de valor a realitzar està dins els límits establerts.
- Conseqüència: si es compleix la condició d'entrada es farà el canvi.
- Prioritat: prioritat d'execució.

A la imatge següent veiem el write_rule:

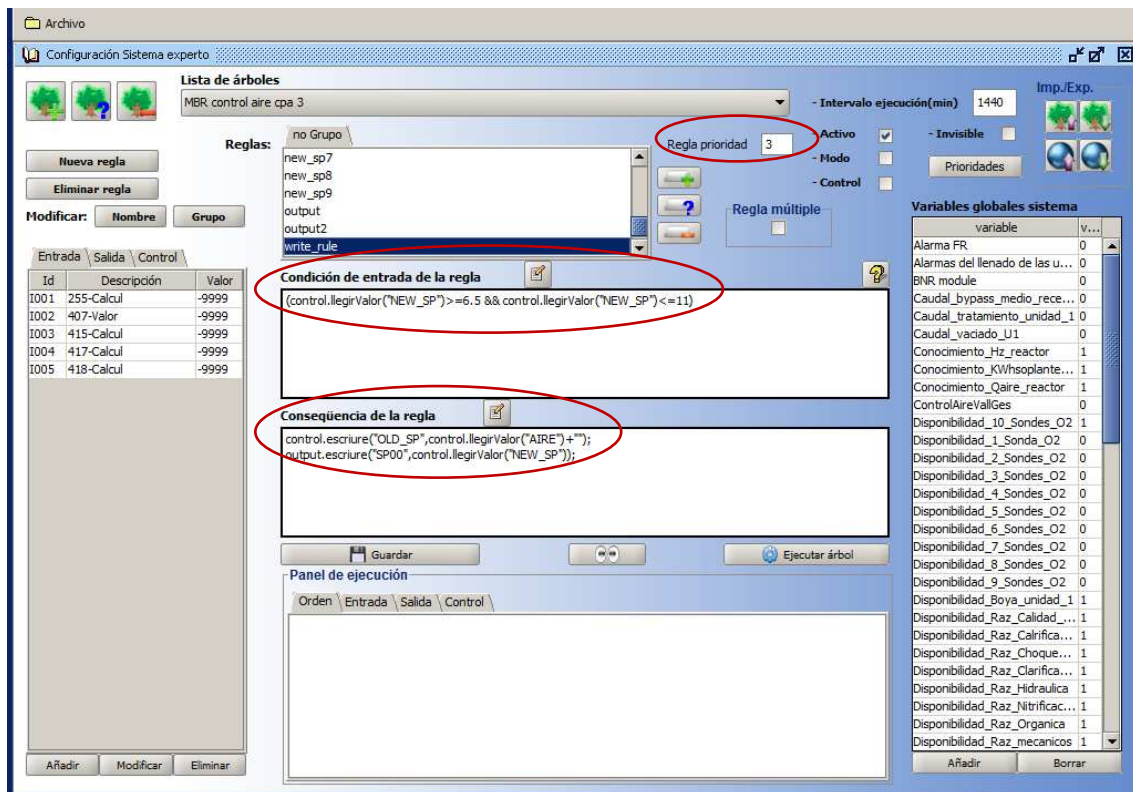


Figura 57. Exemple de write_rule

Sortida:

- Condicó: $0 \geq 0$ perquè sempre ha de donar una sortida.
- Conseqüència: conté la informació que s'ha de mostrar a l'aplicació web.
- Prioritat: prioritat d'execució, serà 0 perquè és el darrer que s'ha d'executar.

Es mostra a continuació la sortida:

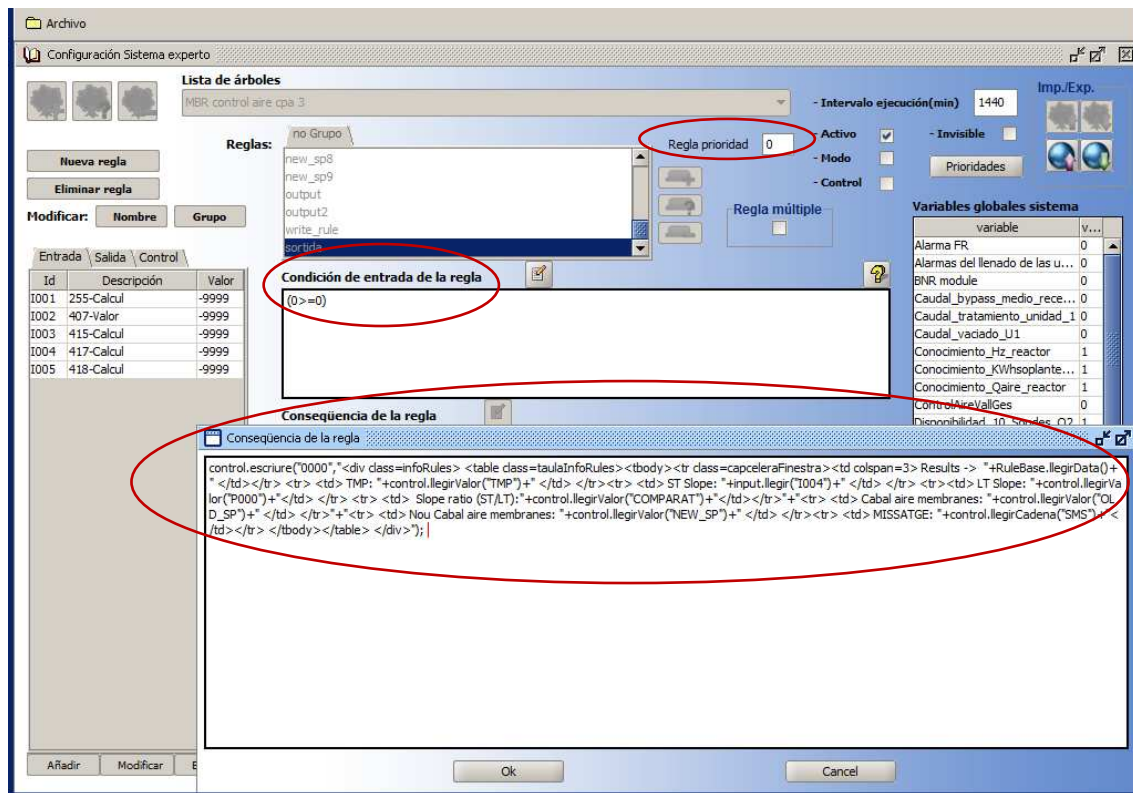


Figura 58. Exemple de sortida

Un cop ho tenim tot implementat cal guardar-ho prement el botó *Guardar*.

Seguidament, per tal d'obtenir el fitxer XML que conté tota la informació que hem implementat, caldrà prémer el botó *Exportar* i seleccionar la ubicació on volem guardar-ho, tal i com es veu a la imatge següent:

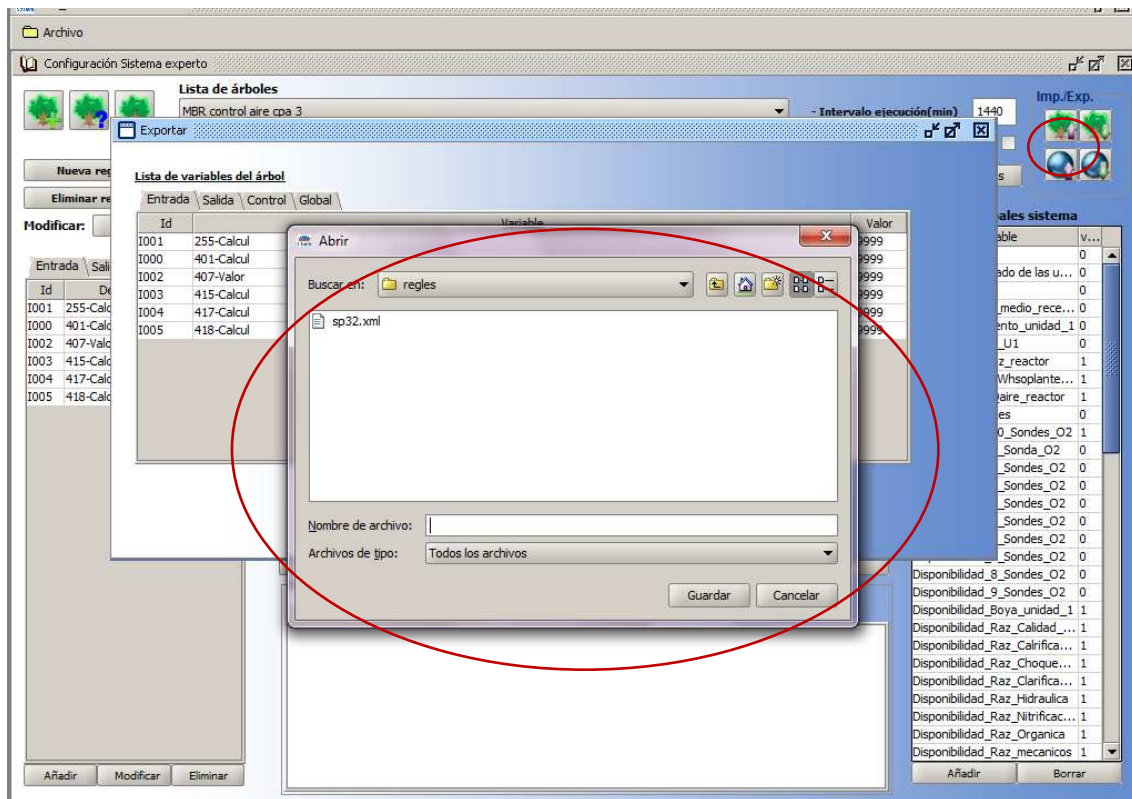


Figura 59. Exportar com a fitxer XML

Podem veure el fitxer XML corresponent a aquesta regla de control al CD adjunt amb el nom *control_aire.pdf*. Aquest codi XML és el que copiarem a la taula *rulebase* de la Base de Dades per tal que l'aplicació *colmatar* pugui llegir-lo, interpretar-lo i executar-lo.

9.1.2. Proves d'execució

Primerament es va comprovar que els càlculs diaris es feien correctament.

Es va executar en local el càlcul en un moment determinat i es va comprovar que es feia correctament. Un cop a escala real, també es va anar comprovant que els càlculs diaris es realitzaven correctament a la planta pilot.

Un cop validats els càlculs diaris i implementada la regla de control, es varen realitzar varies proves d'execució en local per tal de comprovar que actuava correctament segons l'arbre de decisions establert. Les proves es realitzen fent una simulació de les dades que intervenen, amb diferents valors que pot agafar el sensor real i es comprova que les diferents branques de l'arbre actuïn de forma correcta.

9.2. StartUp Kubota Module

Afegir a la classe *MySQLPersistenceHistoValor*, els càlculs diaris que necessitem.

Segons s'acorda a les reunions amb els membres del grup, el càlcul diari a fer serà la mitjana diària de totes les mitjanes de cada cicle que hi hagi durant un dia.

Un cicle es considera uns minuts determinats en què la planta està permeant, seguidament fa un temps de relaxació, així doncs considerem un cicle els minuts en què està permeant, i serà vàlid si aquest dura més de dos minuts.

Per al càlcul diari es fa una detecció del cicle mitjançant la *Bomba de permeat* (aparell que ens permet saber que quan la seva senyal és 1 està permeant i quan és 0 està relaxant), aparell que tenim identificat al nostre sistema amb id 197 i amb una adreça de memòria determinada.

Un cop podem detectar el cicle, se n'agafen totes les dades i es filtren segons un criteri establert per eliminar valors que considerariem no correctes. Un cop fet això es calcula una mitjana de cada cicle. A partir de la mitjana de cada cicle i sabent quants cicles hi ha durant un dia sencer, podrem saber la mitjana diària.

Aquest tipus de càlcul l'utilitzem per a conèixer la *TMP diària* que identifiquem al nostre sistema amb la id 255 (ja calculat a la implementació de la regla de control *MBR Control Aire Membranes*), per conèixer el *Pendent diari de TMP* amb id 356, els *MLSS diaris* amb id 444 i la *Filterability diària* amb id 445.

Un cop tenim totes les dades precalculades que necessitem, havent comprovat que cada dia es calculen aquests paràmetres i un cop verificat que el càlcul és correcte, es pot procedir a dissenyar la regla de control.

9.2.1. Disseny de la regla de control

Per al disseny de la regla de control utilitzem una aplicació ja existent i mencionada anteriorment.

Primerament hem d'identificar quins paràmetres necessitem d'entrada, quins paràmetres de control i quins de sortida.

Per tant tindrem que necessitem d'entrada:

- Consigna cabal de treball: aparell id 104.

- Flux instantani: aparell id 187.
- TMP diària: aparell id 255.
- Pendent TMP diària: aparell id 356.
- MLSS diaris: aparell id 444.
- Filterability diària: aparell id 445.
- Consigna % temps marxa bomba purga: aparell id 442.

Introduïrem els paràmetres d'entrada com es veu a la figura:

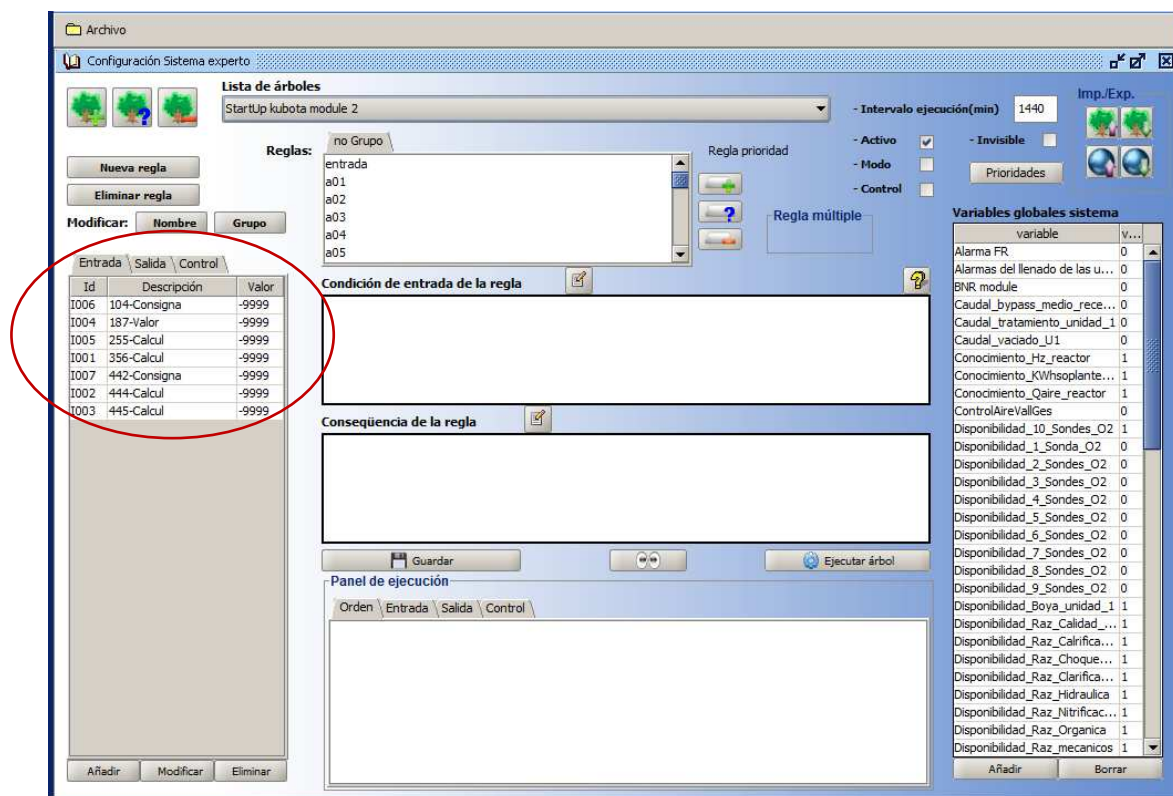


Figura 60. Paràmetres d'Entrada

Com a paràmetres de control necessitarem:

- tmp_high: valor màxim de TMP.
- tmp_low: valor mínim de TMP.
- fr_high: valor màxim de Pendent de TMP.

- fr_low: valor mínim de Pendent de TMP.
- filt_high: valor màxim de Filterability diària.
- filt_low: valor mínim de Filterability diària.
- design_flux: Flux del disseny.
- mlss_high: valor màxim MLSS diaris.
- mlss_low: valor mínim MLSS diaris.
- anoxic: valor per realitzar càlculs.
- anaerobic: valor per realitzar càlculs.
- principal: valor per realitzar càlculs.
- purga: valor per realitzar càlculs.
- segons: valor per realitzar càlculs.
- segonsP: valor per realitzar càlculs.
- minuts: valor per realitzar càlculs.

A la següent imatge es pot veure els paràmetres de control:

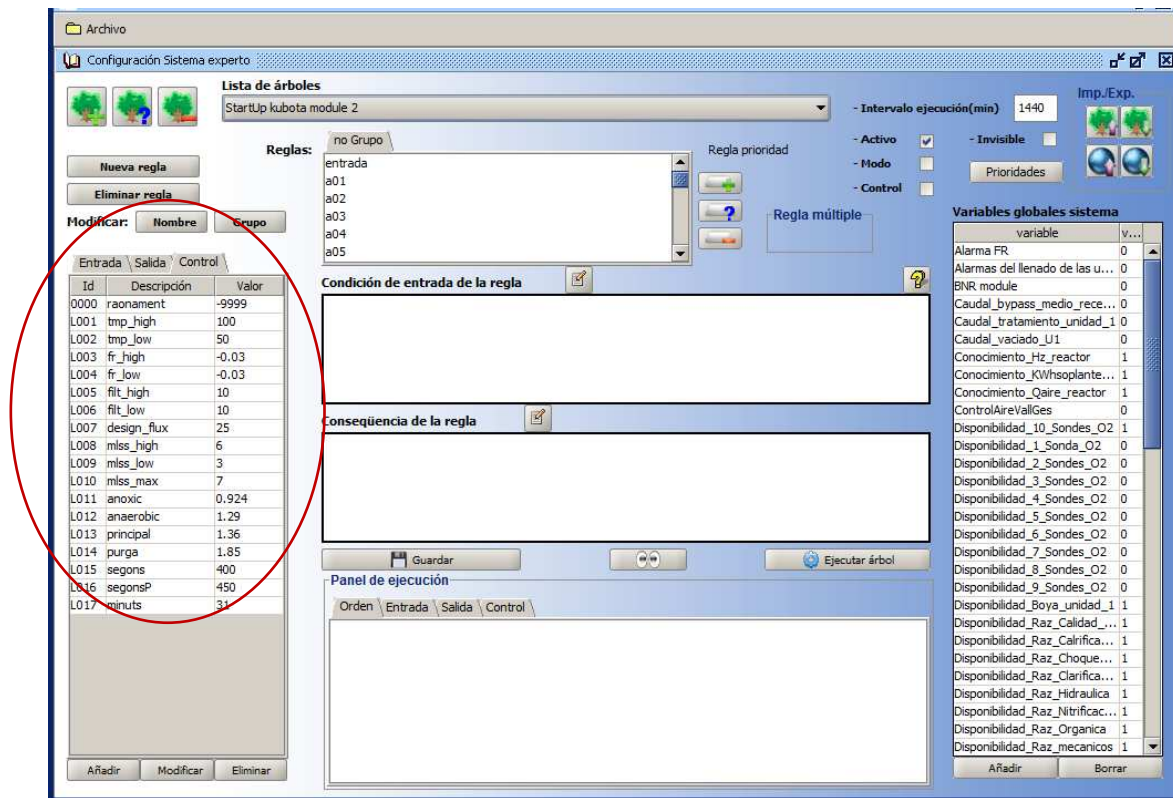


Figura 61. Paràmetres de Control

Paràmetres de sortida:

- Consigna temps aturada bomba purga: aparell id 133.
- Consigna temps marxa bomba purga: aparell id 134.
- Consigna temps aturada recirculadora anòxic: aparell id 126.
- Consigna temps marxa recirculadora anòxic: aparell id 125.
- Consigna temps aturada recirculadora anaeròbic: aparell id 124.
- Consigna temps marxa recirculadora anaeròbic: aparell id 123.
- Consigna temps aturada recirculadora principal: id aparell 122.
- Consigna temps marxa recirculadora principal: id aparell 121.
- Consigna cabal de treball: aparell id 104.

Podem veure els paràmetres de sortida a la imatge següent:

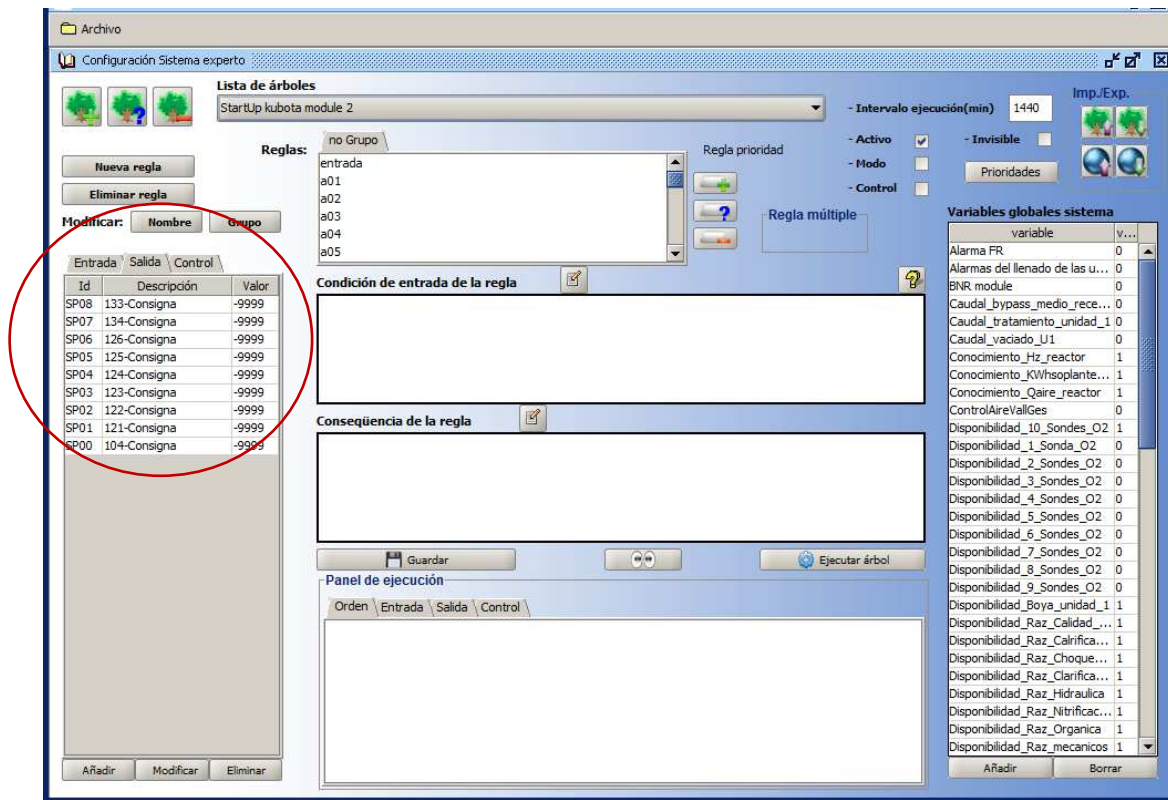


Figura 62. Paràmetres de sortida

Per a la implementació bàsicament tindrem diferents parts a definir:

- entrada: on identificarem cada aparell de l'entrada que hem de conèixer el seu valor per un nom que ens permetrà treballar més fàcilment a la implementació:
- aXX: es definiran totes les branques possibles de la implementació de la regla de control amb les seves condicions d'entrada i la corresponent conseqüència.
- write_ruleX: on es realitza el canvi de valor dels aparells que hem de modificar.
- sortidaX: es defineix tot el que volem que es mostri a l'aplicació web com a resultat de l'execució de la regla de control.

Cada part tindrà un valor per indicar-ne la prioritat de la seva execució.

Entrada:

- Condicció: $0 \geq 0$ perquè volem que entri sempre.

- Conseqüència: llegirà els valors que li donem a Entrada.
- Prioritat: indica la prioritat d'execució.

A la imatge següent podem veure l'entrada:

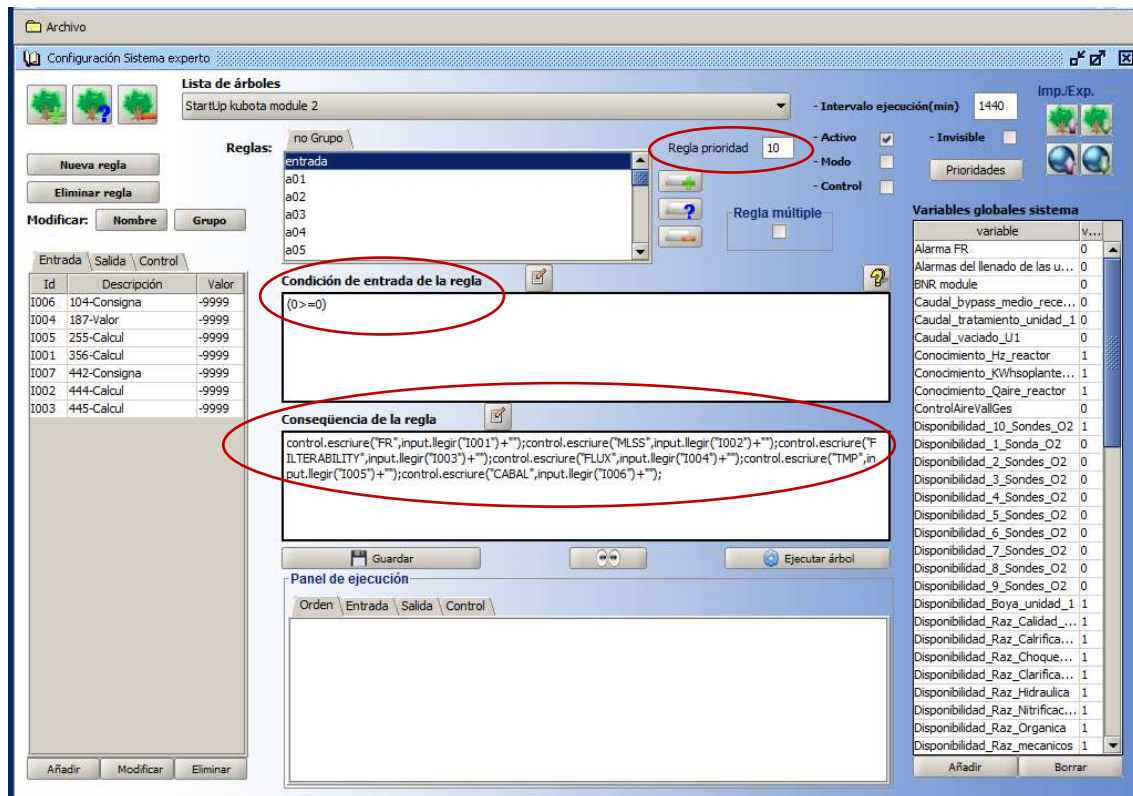


Figura 63. Exemple de la condició d'entrada

aXX:

Exemple d'una de les branques de la regla de control.

- Condició: condicions d'entrada segons els paràmetres llegits.
- Conseqüència: conseqüència si es compleix la condició d'entrada.
- Prioritat: prioritat d'execució.

A la imatge següent es pot veure l'exemple d'una de les branques de l'arbre:

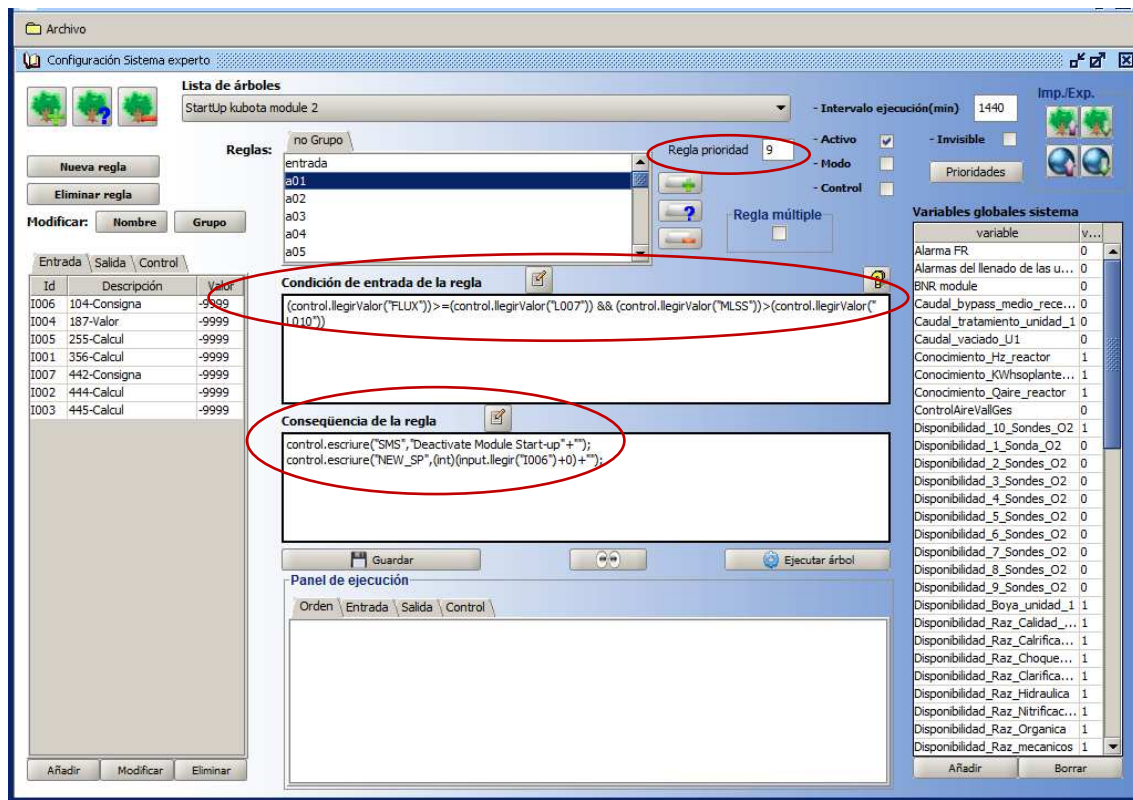


Figura 64. Exemple d'una de les branques

Es mostra a la imatge següent, la branca de l'arbre implementada com a exemple:

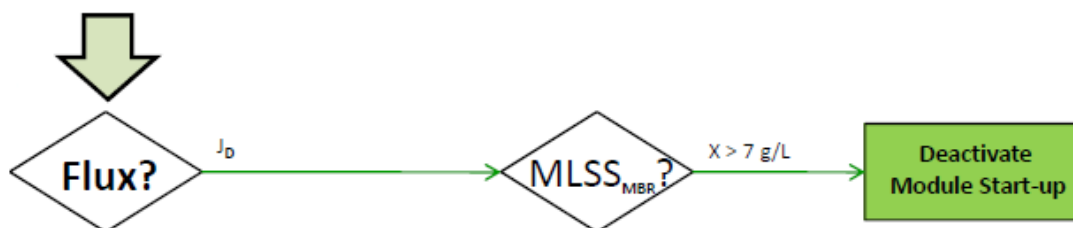


Figura 65. Branca de l'arbre corresponent a l'exemple aXX

Write rule:

- Condicció: el canvi de valor a realitzar compleix unes condicions.
- Conseqüència: si es compleix la condició d'entrada es farà el canvi.
- Prioritat: prioritat d'execució.

Veiem a la imatge que hi ha a continuació un dels write_rule:

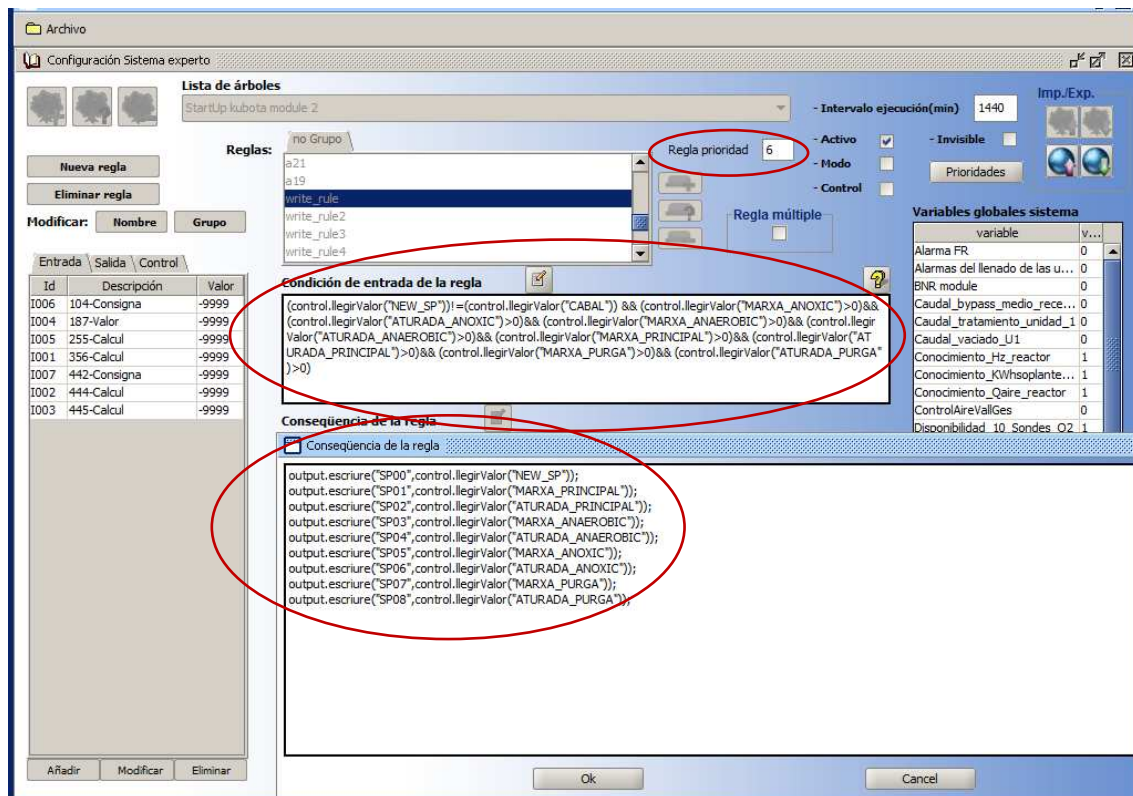


Figura 66. Exemple d'un dels write_rule

Sortida:

- Condicó: Comprovar si hi ha o no canvi.
- Conseqüència: Si es compleix la condició d'entrada, mostrar tota la informació que volem veure a l'aplicació web.
- Prioritat: prioritat d'execució, serà 0 perquè és el darrer que s'ha d'executar.

A la imatge següent podem veure una de les sortides:

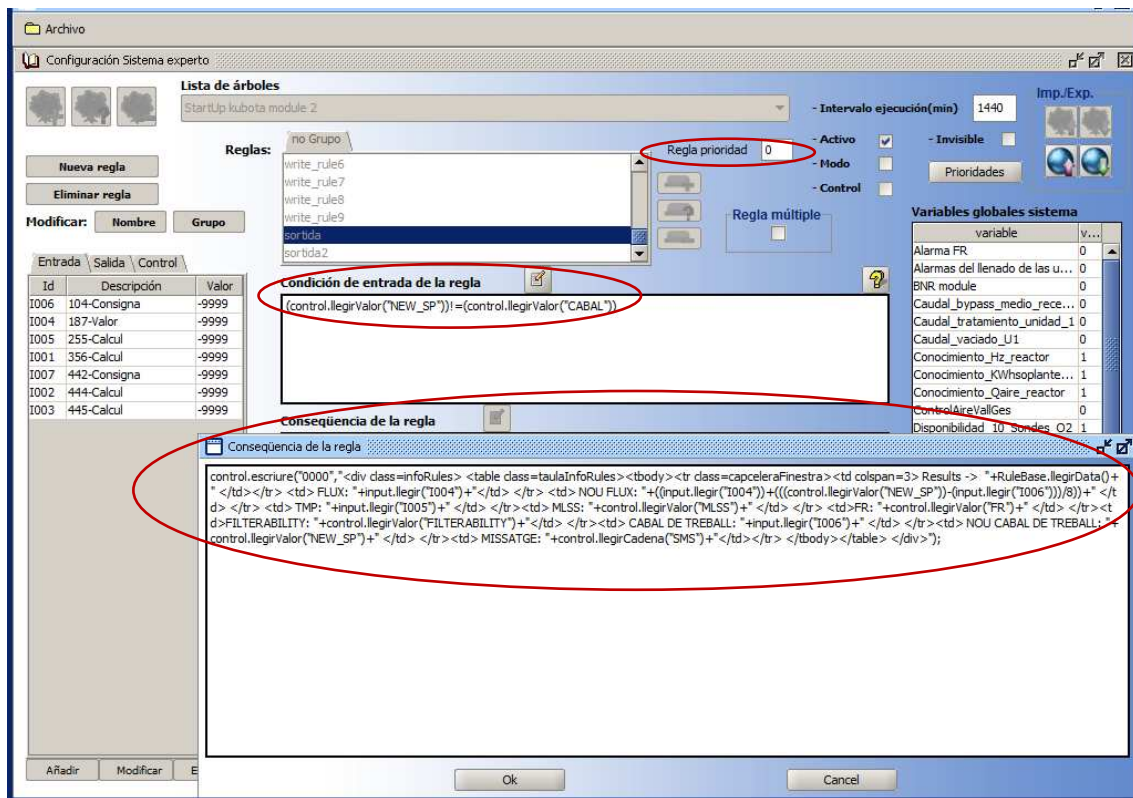


Figura 67. Exemple d'una de les possibles sortides

Un cop ho tenim tot implementat cal guardar-ho prement el botó *Guardar*.

Seguidament, per tal d'obtenir el fitxer XML que conté tota la informació que hem implementat, caldrà prémer el botó *Exportar* i seleccionar la ubicació on volem guardar-ho, tal i com es veu a la imatge que hi ha a continuació:

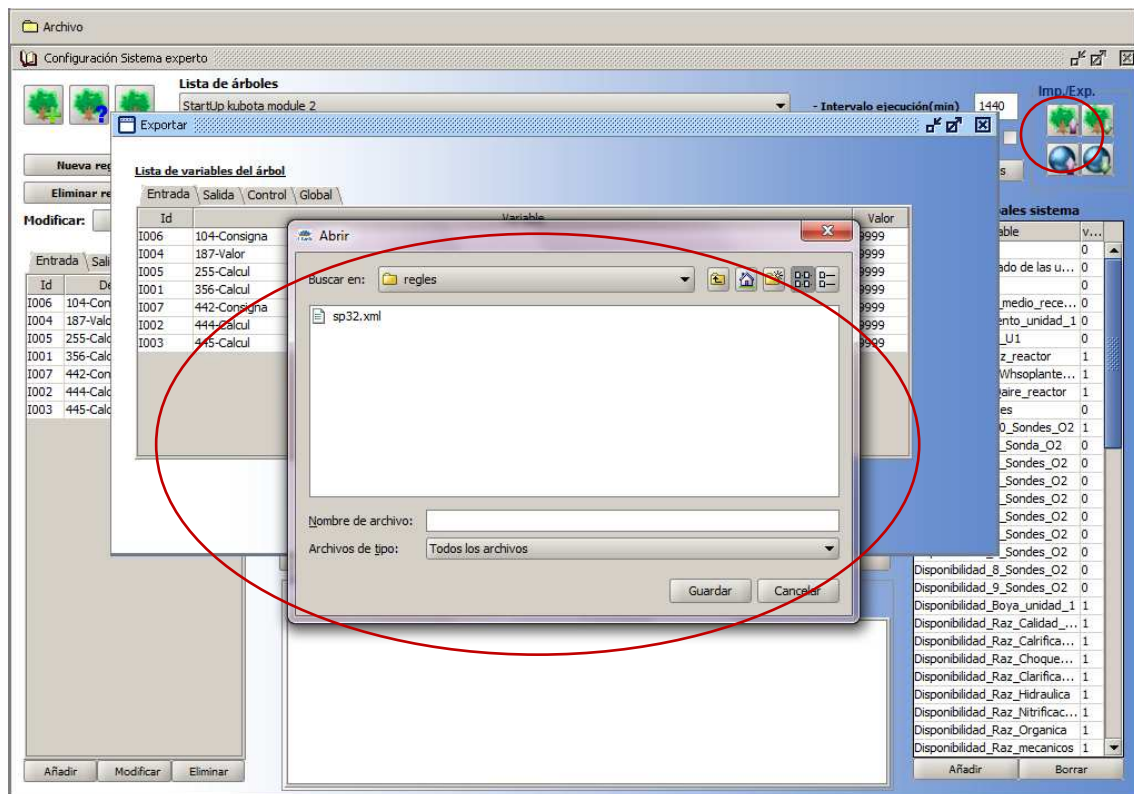


Figura 68. Com exportar el fitxer XML

Podem veure el fitxer XML corresponent a aquesta regla de control al CD adjunt amb el nom *startup.pdf*, aquest codi XML és el que copiarem a la taula *rulebase* de la Base de Dades per tal que l'aplicació *colmatar* pugui llegir-lo, interpretar-lo i executar-lo.

9.2.2. Proves d'execució

Primerament es va comprovar que els càlculs diaris es feien correctament.

Es va executar en local el càlcul en un moment determinat i es va comprovar que es feia correctament. Un cop fet això en escala real, també es va anar comprovant que els càlculs diaris es realitzaven correctament a la planta pilot.

Un cop validats els càlculs diaris i implementada la regla de control, es varen realitzar varies proves d'execució en local per tal de comprovar que actuava correctament segons l'arbre de decisions establert. Les proves es realitzen fent una simulació de les dades que intervenen, amb diferents valors que pot agafar el sensor real i es comprova que les diferents branques de l'arbre actuïn de forma correcte.

10. Implantació i resultats

En aquest apartat s'explicarà la implantació i resultats obtinguts de les dues regles de control implementades en aquest PFC.

Article relacionat:

Joaquim Comas, Hèctor Monclús, Giuliana Ferrero, Ignasi Rodriguez-Roda, Luis Sancho i Eduardo Ayesa (2010). "Development and Validation of a Decision Support System for the Integrated Operation of Membrane Bioreactors". *International Environmental Modelling and Software Society (iEMSs) 2010, Ottawa, Canada*.

que es troba al CD adjunt a aquesta memòria.

10.1. MBR Control Aire Membranes

L'experiment per provar la regla de control MBR Control Aire Membranes implementada es va dur a terme del 08/07/2011 fins al 05/12/2011.

Les variables més importants són la *TMP* i la *Permeabilitat*, tal i com es pot veure a la implementació a l'apartat 8.4.1. Com a variable manipulada hi ha el cabal d'aire. Finalment, en el MBR hi ha instal·lada una sonda de sòlids (MLSS) degut a ser una variable a tenir molt en consideració en membranes *FS*, ja que se'n necessita un valor mínim de $5 \text{ g}\cdot\text{L}^{-1}$ per tal que el fregament superficial permeti una neteja més eficient, que en el sistema informàtic no es té en compte però que per als investigadors és important. Es poden veure els resultats obtinguts en el gràfic següent:

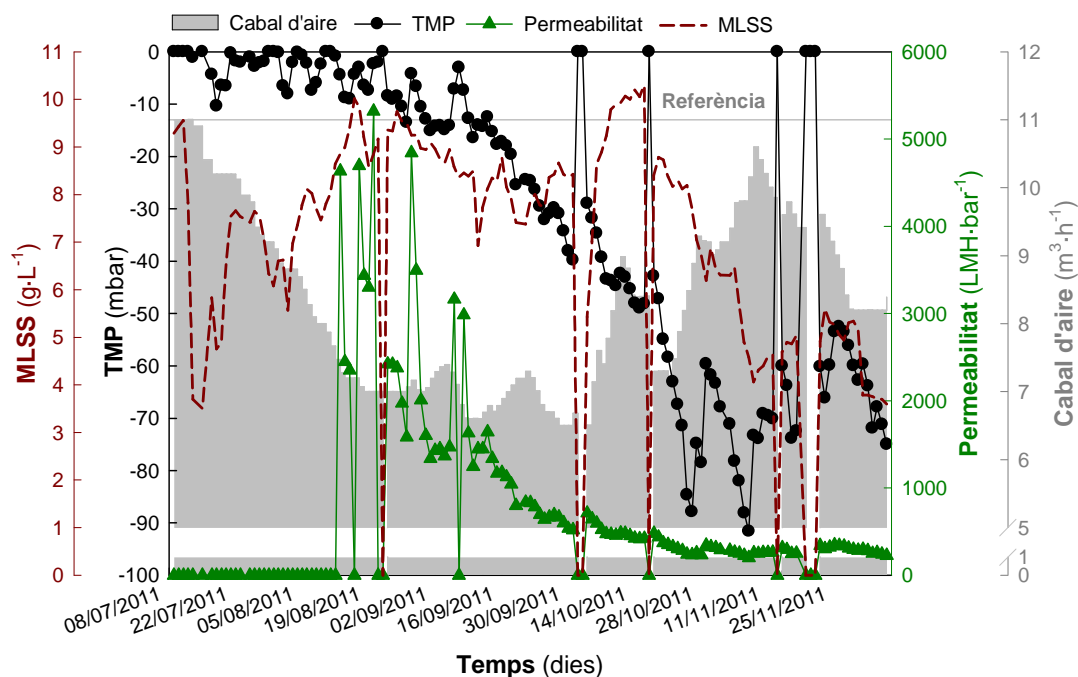


Figura 69. Resultats de l'experimentació en planta pilot des del dia 08/07/2011 fins el 05/12/2011.

Malgrat que alguns dies va haver-hi alguna incidència a la planta, com per exemple que la bomba d'entrada quedava aturada (el sistema d'enviament de correus electrònics va enviar els correus electrònics corresponents), es va aconseguir estalviar aire sense embrutir prematurament les membranes.

Partint del cabal d'aire recomanat pel fabricant ($11 \text{ m}^3 \cdot \text{h}^{-1}$) s'ha arribat als següents valors d'estalvi:

- *Valor mínim d'aire:* $6,5 \text{ m}^3 \cdot \text{h}^{-1}$. Això representa un **41% d'estalvi** respecte el funcionament normal.
- *Valor mitjà d'aire consumit:* $8,25 \text{ m}^3 \cdot \text{h}^{-1}$. Aquest valor representa un **estalvi del 25%** respecte el que recomana el fabricant.

Articles relacionats:

Giuliana Ferrero; Hector Monclús; Luis Sancho; Juan M Garrido; Joaquim Comas and Ignasi Rodriguez-Roda (2011). "A knowledge-based control system for air-scour optimisation in membrane bioreactors". *Water Science and Technology*. 63(9), 2025-2031.

Giuliana Ferrero, Hector Monclús, Gianluigi Buttiglieri, Sara Gabarron, Joaquim Comas and Ignasi Rodriguez-Roda. (2011). "Development of a control algorithm for air-scour reduction in membrane bioreactors for wastewater treatment". *Journal of Chemical Technology & Biotechnology*. 86(6), 784-789.

que podeu consultar a la web www.colmatar.es o bé trobareu al CD adjunt a aquesta memòria.

A partir d'aquest estudi n'ha sorgit una patent (Referència: 200054). Podem veure'n la notícia relacionada "La UdG obté la patent d'un procediment que millora l'eficiència energètica dels MBR" i que es va publicar a la web de la Universitat de Girona www.udg.edu i al CD adjunt a aquesta memòria.

10.2. StartUp Kubota Module

L'experiment per provar la regla de control StartUp Kubota Module implementada es va dur a terme els primers mesos de l'any 2011 (de gener a juny aproximadament).

Les variables més importants són la *TMP*, el *Flux*, la concentració de sòlids *MLSS*, *FR de TMP* i *Filterability* tal i com es pot veure a la implementació a l'apartat 8.4.2. Com a variable manipulada hi ha el *Cabal de treball* que ens permet canviar el *Flux*.

Primerament es va dur a terme un experiment amb la validació manual de l'arbre de decisions. És a dir, la comprovació i modificació manual de cada variable, com podem veure en la següent imatge:

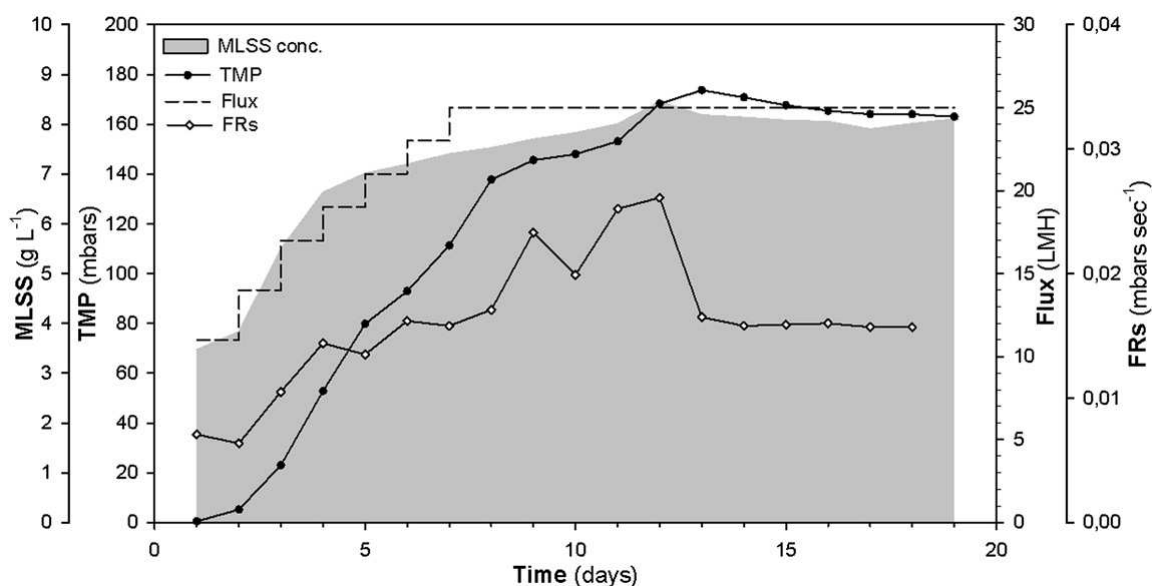


Figura 70. Validació manual de StartUp Kubota Module

Un cop validat, es va procedir a posar en marxa el mòdul de control amb l'automatització d'aquesta regla de control.

Els resultats van ser satisfactoris ja que el sistema automàtic s'ha desenvolupat com una eina per controlar el flux i la concentració de sòlids i minimitzar l'embrutiment BRM en condicions no estacionaries, com ara la posada en marxa d'una planta pilot i l'aplicació de totes les decisions en temps real. S'ha obtingut un sistema automàtic, ràpid i consistent per a la posada en marxa d'una planta pilot i l'assoliment d'una TMP estable i una eliminació de nutrients satisfactòria. Pel cas d'estudi, en només 20 dies, com es mostra en la gràfica següent:

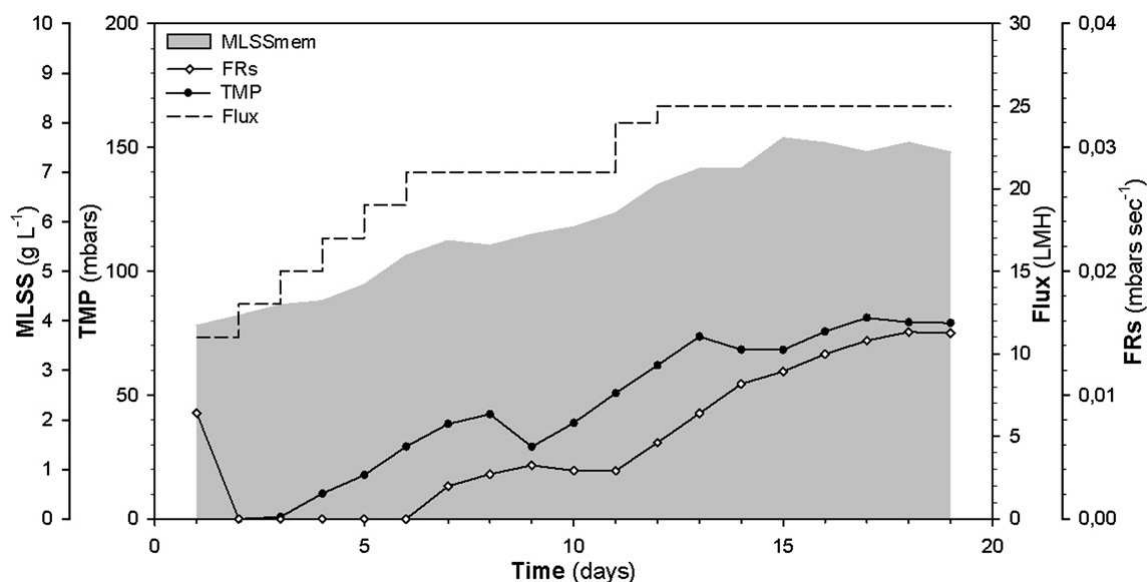


Figura 71. Resultats implantació StartUp Kubota Module

Els resultats i validació d'aquesta regla de control es poden veure en més detall al següent article:

Monclús, H., Buttiglieri, G., Ferrero, G., Rodriguez-Roda, I., and Comas, J. (2012). "Knowledge-based control module for start-up of flat sheet MBRs". *In press at Bioresource Technology*. doi:10.1016/j.biortech.2011.12.001

que podeu consultar a la web www.colmatar.es o bé trobareu al CD adjunt a aquesta memòria.

11. Conclusions

S'han assolit els objectius desitjats d'anàlisi, disseny, implementació i validació d'un SAD intel·ligent pel control remot i la supervisió de bioreactors de membrana.

Aquest SAD és un sistema informàtic que permet l'adquisició i monitorització de les dades i el control robust i autònom de variables, així com la detecció d'errors, alarmes i fallades, alhora que ens permet la supervisió experta i remota dels bioreactors de membrana. El sistema té un registre d'actuacions que posteriorment es pot analitzar i permetre la seva reutilització en el futur. S'ha validat el sistema informàtic en planta pilot a escala industrial.

La meua estada al LEQUIA m'ha proporcionat tant una experiència professional com personal.

Professionalment he après moltes coses, entre elles a desenvolupar eines o parts d'eines que s'utilitzen en experimentació a escala real. També a ser independent a l'hora de buscar noves eines, llenguatges, solucions a problemes que poden sortir, etc. He treballat amb el llenguatge Java, amb JavaScript amb Ajax, amb SQL, entre d'altres.

Personalment he tingut la oportunitat de treballar en un grup molt conegut en el tractament d'aigües. He pogut experimentar com el que he desenvolupat es posa en marxa a escala real i és de gran utilitat a l'hora de realitzar els experiments que els meus companys duen a terme. Sobretot la gran satisfacció d'haver pogut donar suport a dues tesis i d'ajudar o donar suport a la investigació del grup del qual formo part.

La regla de control anomenada "MBR control aire" va donar suport a la tesi de la Dra. Giuliana Ferrero "Development of an air-scour control system for membrane bioreactors".

La regla de control anomenada "StartUp kubota module" va donar suport a la tesi del Dr. Hèctor Monclús "Development of a decision support system for the integrated control of membrane bioreactors".

Podeu consultar les tesis a la web www.colmatar.es o bé al CD adjunt a aquesta memòria.

Amb això vull donar gràcies al grup LEQUIA i sobretot als meus companys de membranes per donar-me la oportunitat de realitzar aquest projecte i tot el que això m'ha aportat, m'aporta i em pugui aportar en un futur.

12. Treball futur

Com a punt de partida el meu PFC *Sistema intel·ligent d'ajuda a la decisió pel control remot de tecnologia de membranes per la depuració d'aigües*, en el que s'han desenvolupat bàsicament dues regles de control, s'està duent a terme un nou projecte que consisteix en la implantació de la regla de control MBR Control Aire a una depuradora a escala real, concretament a la EDAR de La Bisbal d'Empordà.

Tot i que aquest projecte ja queda fora del meu abast he cregut convenient explicar una mica per sobre en què consisteix ja que es un treball futur que ha sorgit de la idea principal d'aquest PFC.

A partir de la implantació a la planta pilot de Castell- Platja d'Aro i després de varies proves de validació, es van realitzar durant cinc mesos on es varen recopilar les dades necessàries i que intervenien en el sistema i també les actuacions que el sistema de control va realitzar sobre la planta. Aquestes dades es varen analitzar per tal de poder obtenir-ne la informació que es necessitava.

Per altra banda es va realitzar l'estudi del funcionament de l'EDAR de La Bisbal, tots els seus components, com actua, com funciona, què hi ha, què es necessita, etc. I que va durar aproximadament un any.

Després d'analitzar les dades obtingudes a la planta pilot de Castell- Platja d'Aro amb el funcionament del control i de l'anàlisi de l'EDAR de La Bisbal es va procedir a dissenyar el sistema de control d'aire adaptat a les condicions d'aquesta EDAR.

Un cop fet això s'ha fet un estudi potencial de l'estalvi energètic i econòmic de l'EDAR.

Finalment s'ha analitzat quina era la inversió que calia fer per tal de poder implementar el Sistema de Control MBR Control Aire a aquella EDAR segons les seves característiques.

A dia d'avui ja s'hi han instal·lat els aparells necessaris per a la implantació del sistema. Els variadors de freqüència per les bufants, analitzadors d'energia elèctrica, tota la part d'adquisició de dades i el sistema informàtic per a implantar el control. Així doncs està tot apunt per quan es cregui convenient posar en marxa aquest Control d'Aire a l'EDAR de La Bisbal.

De tot el possible treball futur podem veure'n un article relacionat de possibles millores:

Ivan Brun, Hèctor Monclús, Jordi Moreno, Montse Dalmau, Ignasi Rodriguez-Roda i Joaquim Comas (2012). “Upgrading a decisión support system for air-scour control in flat sheet membrane bioreactors”.

que també podreu consultar properament a la web www.colmatar.es o bé trobareu al CD adjunt a aquesta memòria.

13. Bibliografia

- Java Platform, Standard Edition 6 – API Specification. 1993, 2011, Oracle and/or its affiliates. <http://docs.oracle.com/javase/6/docs/api/>
- MySQL 5.6 Reference Manual. 1997, 2012, Oracle and/or its affiliates. <http://dev.mysql.com/doc/refman/5.6/en/index.html>
- PostgreSQL. 2009-2011. PostgreSQL-es – Rafael Martinez. <http://www.postgresql.org.es/>
- Instal·lació i configuració Servidor OPC. http://www.infopl.net/files/descargas/omron/infoPLC_net_Tknika_Actividad_7_Configuracion_Cx-Server_OPC_Omron.pdf
- Comas, J. *et al* (2010). “Development and Validation of a Decision Support System for the Integrated Operation of Membrane Bioreactors”.
- Ferrero, G. *et al* (2011). “A knowledge-based control system for air-scour optimisation in membrane bioreactors”. *Water Science and Technology*. 63(9), 2025-2031.
- Ferrero, G. *et al* (2011). “Development of a control algorithm for air-scour reduction in membrane bioreactors for wastewater treatment”. *Journal of Chemical Technology & Biotechnology*. 86(6), 784-789.
- Monclús, H. *et al* (2012). “Knowledge-based control module for start-up of flat sheet MBRs”. *In press at Bioresource Technology*. doi:10.1016/j.biortech.2011.12.001
- Brun, I. *et al* (2012). “Upgrading a decisión support system for air-scour control in flat sheet membrane bioreactors”.
- Hèctor Monclús (2011).” Development of a decision support system for the integrated control of membrane bioreactors”.
- Giuliana Ferrero (2011). “Development of an air-scour control system for membrane bioreactor.

14. Annexos

En el CD ajunt a aquesta memòria hi ha detallats cadascun dels fitxers:

- Fitxer XML MBR Control Aire Membranes (control_aire.pdf)
- Fitxer XML StartUp Kubota Module (startup.pdf)
- Comas, J. *et al* (2010). "Development and Validation of a Decision Support System for the Integrated Operation of Membrane Bioreactors". (Comas.pdf)
- Ferrero, G. *et al* (2011). "A knowledge-based control system for air-scour optimisation in membrane bioreactors". *Water Science and Technology*. 63(9), 2025-2031. (Ferrero_wst.pdf)
- Ferrero, G. *et al* (2011). "Development of a control algorithm for air-scour reduction in membrane bioreactors for wastewater treatment". *Journal of Chemical Technology & Biotechnology*. 86(6), 784-789. (Ferrero_jctb.pdf)
- Monclús, H. *et al* (2012). "Knowledge-based control module for start-up of flat sheet MBRs". *In press at Bioresource Technology*. doi:10.1016/j.biortech.2011.12.001. (Monclus.pdf)
- Brun, I. *et al* (2012). "Upgrading a decisión support system for air-scour control in flat sheet membrane bioreactors". (Brun.pdf)
- Hèctor Monclús (2011). "Development of a decision support system for the integrated control of membrane bioreactors". (Tesi_Monclus.pdf)
- Giuliana Ferrero (2011). "Development of an air-scour control system for membrane bioreactors". (Tesi_Ferrero.pdf)
- Noticia Patent UdG. (patent.pdf)
- Noticia Tesis UdG. (tesis.pdf)

15. Manual d'usuari i/o instal·lació

Diposem d'un PC amb Windows XP on hi instal·lem un servidor OPC que ens proporciona l'instal·lador dels PLC's. En el cas de la planta pilot de Castell d'Aro tenim el servidor OPC CX-Server.

Podem veure un exemple d'instal·lació al document "installacio_opc.pdf" que conté el CD adjunt a aquesta memòria.

Un cop fet això hem de configurar el DCOM per tal que funcioni correctament la comunicació entre els PLC's i el Servidor OPC. Podem veure'n la configuració al document "dcom.pdf" que conté el CD adjunt a aquesta memòria.

Després instal·lem el MySQL Server (segons manual d'instal·lació) que podem trobar a la web <http://dev.mysql.com/doc/refman/5.0/es/installing.html>.

Instal·lem un servidor web Tomcat, que és des d'on podrem posar en marxa l'aplicació web per tal de visualitzar totes les dades.

Amb el codi Java de l'aplicació colmatar, generem un arxiu .jar que executarem per tal que l'aplicació funcioni i reculli les dades que ens venen donades des del PLC.

L'aplicació colmatar i l'aplicació web estan detallades als apartats 8.1 i 8.2 respectivament.