
Aplicació i serveis web per Ichnaea Software

Autor:

Nahuel Velazco Sanchez

Supervisor:

Luis Antonio Belanche Muñoz

Dept. de Llenguatges i Sistemes Informàtics

Enginyeria Informàtica
Facultat d'Informàtica de Barcelona



24 de maig de 2014

DADES DEL PROJECTE

Títol del projecte: Aplicació i serveis webs per Ichnaea Software

Nom de l'estudiant: Nahuel Velazco Sanchez

Titulació: Enginyeria Informàtica

Crèdits: 37,5

Director: Luis Antonio Belanche Muñoz

Departament: Llenguatges i Sistemes Informàtics

MEMBRES DEL TRIBUNAL *(nom i signatura)*

President: Fatos Xhafa

Vocal: Rafael Farrè Cirera

Secretari: Luis Antonio Belanche Muñoz

QUALIFICACIÓ

Qualificació numèrica:

Qualificació descriptiva:

Data:

Índex

1	Prefaci	1
1.1	Abast	1
1.2	Objectius	2
2	Introducció a Ichnaea	3
2.1	MST: Microbial Source Tracking	3
2.2	Ichnaea Software	3
2.3	L'univers d'Ichnaea	4
2.3.1	Matrius	4
2.3.2	La matriu i les mostres	4
2.3.3	Entrenaments	5
2.3.4	Matrius de prediccions	5
2.3.5	Sistema de cues	5
3	Anàlisis i Especificació	7
3.1	Anàlisis de Requeriments	7
3.1.1	Requeriments funcionals	7
3.1.2	Requeriments no funcionals	9
3.2	Model de Casos d'ús	9
3.2.1	Actors	9
3.2.2	Diagrama dels casos d'ús	10
3.2.3	Especificació dels casos d'ús	10
3.3	Model Conceptual	32
3.3.1	Diagrama de classes	32
3.3.2	Explicació de les classes	32
3.4	Model d'estats	33
3.4.1	Estats dels entrenaments	34
3.4.2	Estats de les prediccions	35
3.5	Model del comportament	35

4	Disseny	45
4.1	Esquema general lògic arquitectònic del sistema	45
4.2	Patró de disseny	46
4.2.1	Esquema del disseny	46
4.2.2	Explicació del disseny	47
4.3	Disseny d'interfícies	48
4.3.1	Disseny principal	48
4.3.2	Interfície de configuració de matrius	49
5	Implementació	53
5.1	Estudi previ de les tecnologies	53
5.1.1	<i>Framework</i> a mida	53
5.1.2	<i>Codeigniter</i>	53
5.1.3	<i>Symfony2</i>	54
5.2	Implementació	54
5.2.1	<i>Symfony2</i>	55
5.2.2	Gestió de dependències	55
5.2.3	Recursos	56
5.3	API: llibreria de serveis web	56
5.4	Integració amb el sistema de cues RabbitMQP	57
5.4.1	Introducció a l'arquitectura de cues: AMQP	57
5.4.2	Consumidors i gestió de resultats	58
6	Proves i tests	61
6.1	Prova en PC local	61
6.2	Prova en entorn distribuït	63
6.3	Erroris coneguts	64
7	Conclusions	65
7.1	Metodologia àgil	65
7.1.1	Backlog	65
7.2	Estimació econòmica	66
7.3	Millores en futures versions	66
8	Bibliografia	1
	Bibliografia	3
A	Manual d'usuari	5
A.1	Casa de l'usuari	5
A.1.1	Llistat dels meus trainings pendants	5
A.2	Variables	6

A.2.1	Veure les variables del sistema	6
A.2.2	Formulari de edició d'una variable	6
A.2.3	Crear una "season set" d'una variable	7
A.2.4	Editar una "Season Set" d'una variable	7
A.2.5	Afegir una "season" a una "season set" existent d'una variable	8
A.3	Matrius	8
A.3.1	Crear una matriu desde un fitxer csv o excel	8
A.4	Configurar una matriu	9
A.4.1	Alies de una columna	10
A.4.2	Especificar en una columna una variable i un conjunt de fitxers	10
A.4.3	Cambiar la visualització	11
A.4.4	Visualitzar missatges	11
A.4.5	Actualitzar una dada d'una fila i d'una columna	11
A.5	Clonar una matriu	11
A.6	Crear un training d'una matriu	11
A.6.1	Simular un training de la matriu Cyprus	13
A.7	Visualitzar un training	13
A.7.1	Problemàtiques de la creació de trainings	13
A.8	Crear una matriu de predicció	14
A.9	Crear una predicció	14
A.10	Actualitzar una matriu de predicció	15
A.11	Llistar les meves prediccions	15
A.12	Visualitzar una matriu de predicció	15
A.12.1	Actualitzar una matriu de predicció	15
A.12.2	Executar una predicció de una matriu de predicció . . .	15
B	Manual d'administrador	17
B.1	Llistar usuaris del sistema	17
B.2	Canviar el grup de l'usuari	17
B.3	Comprovar cua	17

Capítol 1

Prefaci

1.1 Abast

L'objectiu del projecte es desenvolupar un conjunts de serveis web per manejar l'algoritme de Backtracking bacteriològic Ichnaea. Actualment Ichnaea es troba en la versió 2.0, desenvolupat per Aitor Perez Pérez Pérez. La primera versió va ser desenvolupada per David Sánchez.

La complexitat de les entrades i configuracions dels paràmetres de entrada de Ichnaea, requereixen de unes interfícies enriquides i d'un model de dades flexible per poder executar l'algoritme. Per un altre banda i com a segon objectiu principal és integrar el Projecte de Final de Carrera de Miguel Ibero "Sistema de cues per Ichnaea Software", on s'està dissenyant i desenvolupant un sistema de cues per manejar les execucions; Ichnaea requereixen d'un temps elevat de procés. Com a tercer objectiu, s'ha de dissenyar e implementar aquest sistema en un entorn distribuït en xarxa.

El document següent conte:

- En el capítol 2 es fa una petita introducció a Ichnaea i al problema MST.
- En el capítol 3 s'especifica els requeriments, els casos d'usos i el model de dades.
- En el capítol 4 s'especifica el disseny de l'aplicació.
- En el capítol 5 es descriuen les tecnologies i la implementació del projecte.
- En el capítol 6 es descriuen les proves i les dificultats trobades.

- En el capítol 7 s'especifica l'estudi econòmic del projecte i les possibles millores.
- En l'annex A es dona un petit manual d'usuari.
- En l'annex B es dona un petit manual de administrador.

1.2 Objectius

Els objectiu principals del projecte son:

- Especificar e implementar les interfícies de usuari per poder configurar les entrades i execució del software Ichnaea.
- Especificar e implementar interfícies d'usuari per poder veure els resultats de la execució del software Ichnaea.
- Interfícies usables, comprensibles i enriquides per tenir una bona experiència de usuari.
- Prototipus de llibreria API per en un futur escalar-la i poder integrar el projecte amb qualsevol perifèric o tecnologia.
- Implementar tots aquests objectius en una tecnologia distribuïda en xarxa.
- Dissenyar un model de dades flexible que permeti relacionar els objectes per a futures versions de Ichnaea i noves funcionalitats que es puguin desenvolupar mitjançant modificacions o millores de les interfícies.
- Integrar la aplicació web amb el projecte "Sistema de cues per Ichnaea Software" de Miguel Ibero.

Capítol 2

Introducció a Ichnaea

En aquest capítol es descriu breument l'univers MST e Ichnaea, necessari per entendre el requeriments. No es dona una visió completa del *software* de com funciona, sino una visió global del seu objectiu i quins elements utilitza.

2.1 MST: Microbial Source Tracking

MST és un problema obert en l'actualitat. Consisteix en determinar l'origen biològic dels residus fecals en cossos aquosos mitjançant l'ús d'indicadors químics i microbiòlegs [2]. Per fer això es prenen mostres i s'analitzen en un laboratori, i segons els resultats, es decideix si contenen residus fecals d'origen humà o de quina família de animals [1].

Prendre aquesta decisió és molt difícil. Fins i tot, els microbiòlegs no estan completament segurs de determinar la font d'infecció de les mostres d'aigües contaminades. La raó es que les mostres son extretes directament de l'entorn i per això estan diluïdes i envellides [1].

L'estudi de l'origen de la pol·lució en cossos aquosos és un problema gran i pot ajudar a assegurar la protecció de les poblacions humanes, mostrant una varietat d'enfermetats, especialment en països subdesenvolupats [1].

2.2 Ichnaea Software

Ichnaea és un software desenvolupat per ajudar a resoldre el problema MST. És un eina per llegir matrius de dades(mostres mesurades) i construir diversos conjunts de models. Amb l'ajuda d'aquestes bosses de models, pot llegir

noves mostres i fer prediccions dels orígens d'aquestes [1].

Actualment es troba en la versió 2.0. La primera versió va ser desenvolupada, com a Master Thesis per David Sànchez, va donar un primer enfoc al problema MST. La segona versió ha sigut desenvolupada com a Projecte de Final de Carrera per Aitor Pérez Pérez. Ambdues versions han sigut supervisades per Lluís Belanche. Desde la primera versió s'ha refactoritzat el codi i millorat tant el rendiment com els algorismes.

2.3 L'univers d'Ichnaea

A continuació veurem les entitats amb les que treballa Ichnaea per tal poder donar una visió de les dades.

2.3.1 Matrius

Ichnaea processa inicialment unes matrius on és defineixen les mostres de dades extretes, on cada columna representa una variable i cada fila representa una mostra.

Variables i conjunts de envelliments

Les variables de Ichnaea tenen associades uns fitxers. En aquest fitxers s'especifiquen dades mesurades que representen els envelliments de les mostres d'aquestes variables segons la estació de l'any. Aquests fitxers s'agrupen en un conjunt per formar en conjunts d'envelliments.

L'objectiu de tenir diferents conjunts d'envelliments és tenir-ho agrupats els fitxers segons les localitzacions. Per exemple, podem tenir el bacteri *Fecal Coliform* amb dos conjunts de envelliments de dos localitzacions diferents del mon. Per exemple, un conjunt de envelliments pot correspondre a mesures fetes a Nairobi i unes altres a Moscú. I cada fitxer representa una estació de l'any, ja que segons la estació i la localització els envelliments son diferents.

2.3.2 La matriu i les mostres

Cada columna d'aquesta matriu, representa una variable de la matriu. Aquesta variable pot ser:

- Una variable que representa una variable de Ichnaea: "variable single"
- Una variable derivada. Son dos "variables single" relacionades per una operació.
- Una variable d'origen, obligatòria per cada mostra. És una etiqueta que identifica l'origen de la mostra.

Les variables d'origen representa una etiqueta de la mostra per tal de identificar origen de la pol·lució. En aquestes matrius els orígens son obligatoris i cada columna ha de tenir un valor definit.

2.3.3 Entrenaments

Ichnaea processa aquestes matrius amb un conjunt d'entrellaments per calcular una bossa de models. Aquesta procés s'anomena entrenament.

Aquestes bosses de models resultants s'utilitzen per fer prediccions.

2.3.4 Matrius de prediccions

Les dades que necessita Ichnaea per fer prediccions son un conjunt de noves mostres en forma de matriu. A partir d'un entrenament, pot fer prediccions d'orígens de contaminació. Aquestes matrius son molt similars descrites a la secció 2.3.1. La diferència és que les mostres no han de ser completes. Per exemple, les mostres no tenen perquè tenir un origen o poden valors per variables sense definir.

2.3.5 Sistema de cues

Ichnaea Software requereix d'un cost alt de procés tant en rendiment com en temps. Aquest projecte s'ha desenvolupat en paral·lel amb el Projecte de Final de Carrera de Miguel Ibero que implementa un sistema de cues d'execució de Ichnaea. Aquest projecte s'ha de integrar amb aquest sistema seguint els requeriments del projecte "Sistema de cues per a Ichnaea Software".

Capítol 3

Anàlisi i Especificació

En aquest capítol es descriuen els requeriments, les operatives que es necessiten i el model de dades.

3.1 Anàlisi de Requeriments

3.1.1 Requeriments funcionals

Administració d'usuaris

La aplicació ha de estar protegida i autoritzada pels usuaris. Els usuaris autenticats han de tenir permisos i pertànyer a grups amb rols autoritzats per fer certes accions. És una aplicació distribuïda per tant s'ha de dissenyar un sistema que permeti:

- Crear comptes d'usuari.
- Atendre peticions de restaurar contrasenyes.
- Enviar mails de confirmacions de accions.
- Canviar permisos a usuaris.

Administració de variables de sistema

La aplicació ha de gestionar les variables que Ichnaea que Ichnaea utilitza al sistema per poder utilitzar-les amb el *software*. Aquestes variables han de tenir associades conjunts de fitxers.2.3.1) La aplicació ha de poder gestionar les variables, els continguts dels fitxers i les associacions entre les variables i els conjunts de fitxers.

Administració de matrius

La aplicació ha de gestionar i configurar matrius de dades. Per la creació de matrius ha de poder llegir una fulla de càlcul i crear un model de dades que representi una matriu a partir de les dades proporcionades.

La aplicació ha de permetre la configuració de les matrius(mirar 2.3.1):

- Gestionar matrius: crear, actualitzar, esborrar i configurar-les.
- Configurar les columnes d'una matriu com una variable i un conjunt de fitxers en cas de que tinguin més.
- Configurar l'origen d'una mostra de la matriu.
- Configurar la data d'una mostra de la matriu.
- Configurar o actualitzar dades bàsiques d'una matriu.

Administració de entrenaments

El sistema ha de gestionar i crear entrenaments i enviar-ho contra una cua d'execució d'Ichnaea:

- Gestionar entrenaments: crear, configurar i esborrar.
- Enviar les matrius a entrenar a la cua d'execucions d'entrenaments.
- Llegir e interpretar el estat del procés.
- Guardar els resultats dels entrenaments.
- Gestionar les sortides de les execucions.

Administració de matrius de predicció

El sistema ha de gestionar i crear noves matrius de prediccions i enviar-ho contra una cua d'execució.

- Gestionar les matrius de prediccions.
- Enviar a les matrius de prediccions a la cua d'execucions de prediccions.
- Llegir e interpretar el estat del procés.
- Llegir els resultats de les prediccions.
- Gestionar les sortides de les execucions.

3.1.2 Requeriments no funcionals

Els requeriments no funcionals son:

- Bon rendiment
- Escalabilitat
- Mantenible
- Flexibilitat
- Usabilitat

3.2 Model de Casos d'us

3.2.1 Actors

Els usuaris del sistema son:

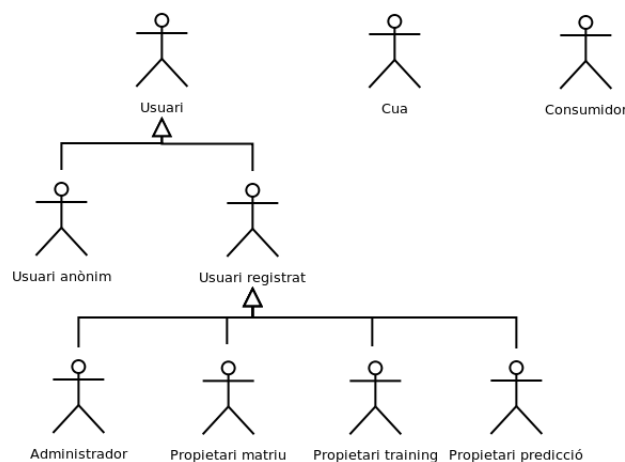


Figura 3.1: Actors

- Usuari anònim: usuari sense compte al sistema.
- Usuari registrat: usuari amb compte al sistema.
 - Propietari d'una matriu: usuari que ha creat una matriu.
 - Propietari d'un entrenament: usuari que ha creat un entrenament.
 - Propietari d'una predicció: usuair que ha creat una matriu de predicció.

- Usuari administrador: usuari amb permisos administratius.
- Cua: usuari lògic(no és una persona física) dels sistema que gestiona les execucions.
- Consumidor: usuari lògic(no és una persona física) del sistema que gestiona les sortides de les execucions.
- Sistema: sistema que rep les peticions i gestiona les sortides.

3.2.2 Diagrama dels casos d'ús

A la figura 3.2 es pot veure la el diagrama de casos d'ús.

3.2.3 Especificació dels casos d'ús

En la documentació s'utilitzarà la següent estructura per definir els casos d'ús:

Identificador	Nom cas d'us
<i>Actors:</i>	Llista de actors
<i>Curs típic d'esdeveniments:</i>	
	1. Esdeveniment
	2. Esdeveniment
	3. ...
<i>Cursos alternatius:</i>	
	1. Esdeveniment Alternatiu
Usuari001	Crear un usuari
<i>Actors:</i>	Anònim

Curs típic d'esdeveniments:

1. Usuari accedeix al formulari de registració. L'usuari introdueix un nom d'usuari, un correu electrònic i una contrasenya per duplicat
2. El sistema envia al usuari una confirmació via correu electrònic amb un enllaç de confirmació i crea un compte no validada.
3. L'usuari accedeix mitjançant l'enllaç de confirmació.
4. El sistema comprova que és un enllaç de confirmació vàlid d'aquest usuari i activa la compte. L'usuari ja està autenticat com usuari bàsic del sistema.

Cursos alternatius:

- 3 El sistema valida que no existeixi un usuari amb aquesta compte de correu i que el correu sigui vàlid. Sino és correcte li informa a l'usuari al mateix formulari.

Usuari002 Canviar un usuari de grup

Actors: Administrador

Curs típic d'esdeveniments:

1. L'administrador llista tots els usuaris del sistema i selecciona un.
2. El sistema mostra un formulari de edició de permisos.
3. L'administrador selecciona el nou permís i confirma l'acció.
4. El sistema guarda els canvis i notifica a l'administrador.

Variable001 Crear una variable

Actors: Usuari administrador

Curs típic d'esdeveniments:

1. L'usuari accedeix a un formulari de creacio.
 2. El sistema mostra un formulari de creacio de variables.
 3. L'usuari dona un identificador i una descripció.
 4. El sistema crea la variable amb la informació donada i confirma a l'usuari.
-

Cursos alternatius:

- 4 El sistema valida que existeixi ja una variable amb aquest identificador i li notifica a l'usuari.
-

Variable002 Actualitzar una variable

Actors: Usuari administrador

Curs típic d'esdeveniments:

1. L'administrador selecciona una variable d'un llistat de variables.
 2. El sistema mostra un formulari d'edició on pot veure els conjunts dels fitxers i pot actualitzar la descripció.
 3. L'usuari modifica la descripció i salva els canvis.
 4. El sistema guarda les modificacions.
-

Variable003 Crear un conjunt de fitxers d'envelliment per una variable

Actors: Usuari registrat

Curs típic d'esdeveniments:

1. L'usuari selecciona una variable d'un llistat de variables.
2. El sistema mostra un formulari d'edició de la variable amb un enllaç a un formulari de creació de conjunt de fitxers.
3. L'usuari accedeix a un formulari de creació d'un conjunt de fitxers.
4. El sistema li mostra un formulari de creació.
5. L'usuari pot donar un nom i seleccionar 0, 1 o 2 fitxers, on cada fitxer pot ser configurat com:
 - a únic per tot l'any
 - com estiu
 - com hivern
 - com tardor
 - com estiu
6. L'usuari salva els canvis.
7. El sistema guarda els canvis i notifica a l'usuari.

Variable004	Actualitzar un conjunt de fitxers d'envelliments
--------------------	---

<i>Actors:</i>	Usuari registrat
----------------	------------------

Curs típic d'esdeveniments:

1. L'usuari selecciona una variable d'un llistat de variables.
2. El sistema li mostra un llistat dels conjunts de fitxers de la variable.
3. L'usuari selecciona un conjunt de fitxers.
4. El sistema li mostra un formulari d'edició del conjunt de fitxers on es pot:
 - Canviar el nom del conjunt de fitxers
 - Esborrar un fitxer
 - Afegir més fitxer i configurar-los com estiu, hivern, tardor, primavera o com tot l'any.
5. L'usuari salva els can.
6. El sistema guarda els canvis i notifica a l'usuari.

Variable005 Esborrar un conjunt d'envelliments d'una variable

Actors: Usuari registrat

Curs típic d'esdeveniments:

1. L'usuari selecciona una variable.
2. El sistema mostra un llistat de conjunts de fitxers per esborrar.
3. L'usuari selecciona del llistat un conjunt de fitxers per esborrar.
4. El sistema li mostra un vista de confirmació de la acció.
5. L'usuari confirma la acció.
6. El sistema esborra tots els fitxers i el conjunt.

Cursos alternatius:

- 4 L'usuari cancel·la la acció.

Variable006 Afegir un nou fitxer a un conjunt de fitxers d'envelliments

Actors: Usuari registrat

Curs típic d'esdeveniments:

1. L'usuari selecciona una variable d'un llistat de variables.
2. El sistema mostra un llistat de conjunts de fitxers de la variable.
3. L'usuari selecciona un conjunt de fitxers.
4. El sistema mostra un formulari de edició del conjunt de fitxers.
5. L'usuari pot seleccionar 0, 1 o 2 fitxers, on cada fitxer pot ser configurat com:
 - a únic per tot l'any
 - com estiu
 - com hivern
 - com tardor
 - com estiu
6. L'usuari salva els canvis.
7. El sistema guarda els canvis.

Variable007 Esborrar un fitxer d'un conjunt de fitxers d'envelliments

Actors: Usuari registrat

Curs típic d'esdeveniments:

1. L'usuari selecciona una variable d'un llistat de variables.
2. El sistema mostra un llistat de conjunts de fitxers de la variable.
3. L'usuari selecciona un conjunt de fitxers.
4. El sistema mostra un llistat dels fitxers.
5. L'usuari selecciona un fitxer per esborrar.
6. El sistema li demana confirmació.
7. L'usuari confirma l'acció.
8. El sistema esborra el fitxer.

Cursos alternatius:

- 6 L'usuari cancel·la la acció.
-

Matriu001 Crear una matriu des d'un fitxer

Actors: Usuari registrat

Curs típic d'esdeveniments:

1. L'usuari accedeix a un formulari de creació de matrius.
2. El sistema mostra el formulari on pot donar nom a la matriu i seleccionar el fitxer en format CSV o Microsoft Excel. L'usuari accepta el formulari.
3. El sistema crear la matriu on:
 - La primera fila del fitxer s'associa com una variable Ichnaea. Si la variable es igual al identificador de la variable, automàticament s'assigna a aquesta columna a aquesta variable i a un conjunt de fitxers d'envelliments per defecte. Les dues ultimes columnes son optatives, on s'especifica l'origen i la data de la mostra.
 - Cada fila del fitxer, després de la primera fila:
 - La primera columna és l'identificador de la mostra. Si conté un alias d'origen, automàticament s'assigna un origen
 - Les columnes restants s'assignen com a valors de la mostra
 - Les dos ultimes columnes seran l'origen o la data segons la primera fila del fitxer.
 - L'usuari visualitza la matriu.

Matriu002 Actualitzar una matriu

Actors: Propietari de la matriu

Curs típic d'esdeveniments:

1. L'usuari selecciona una matriu per actualitzar.
2. El sistema li mostra un formulari d'edició de la matriu.
3. L'usuari accedeix a un formulari de importació.
4. El sistema mostra el formulari d'edició.
5. L'usuari pot actualitzar el nom a la matriu i/o seleccionar el fitxer de tipus de fulla de càlcul. L'usuari confirma els canvis
6. El sistema esborra la matriu anterior i torna a crear la matriu on:
 - La primera fila del fitxer s'associa com una variable Ichnaea. Si la variable es igual al identificador de la variable, automàticament s'assigna a aquesta columna a aquesta variable i a un conjunt de fitxers d'envelliments per defecte. Les dues últimes columnes son optatives, on s'especifica l'origen i la data de la mostra.
 - Cada fila del fitxer, després de la primera fila:
 - La primera columna és l'identificador de la mostra. Si conté un alias d'origen, automàticament s'assigna un origen
 - Les columnes restants s'assignen com a valors de la mostra
 - Les dos últimes columnes seran l'origen o la data segons la primera fila del fitxer.
7. L'usuari visualitza la matriu.

Matriu003 Clonar una matriu

Actors: Usuari registrat

Curs típic d'esdeveniments:

1. L'usuari selecciona una matriu per clonar.
2. El sistema mostra un formulari amb un nom suggerit per la matriu.
3. L'usuari pot canviar el nom i acceptar la clonació.
4. El sistema clona la matriu i la seva configuració sense copiar entrenaments ni prediccions. El propietari de la matriu és l'usuari que ha realitzat la clonació.
5. L'usuari veu la matriu clonada.

Matriu004 Esborrar una matriu

Actors: Propietari de la matriu

Curs típic d'esdeveniments:

1. L'usuari selecciona una matriu del sistema per esborrar.
2. El sistema demana confirmació per esborrar la matriu.
3. L'usuari confirma la acció.
4. El sistema clona la matriu i la seva configuració sense copiar entrenaments ni prediccions. El propietari de la matriu és l'usuari que ha realitzat la clonació.

Cursos alternatius:

- 3 L'usuari cancel·la la acció.
-

Matriu005 Configurar la columna d'una matriu

Actors: Usuari propietari de la matriu

Curs típic d'esdeveniments:

1. L'usuari selecciona una matriu
 2. El sistema mostra una vista per configurar les columnes de una matriu.
 3. L'usuari selecciona una columna i pot:
 - donar un nom a la columna
 - seleccionar una variable i una conjunt d'envelliments de la variable
 4. L'usuari accepta la configuració
 5. El sistema salva la configuració de la columna
-

Matriu006 Configurar una mostra d'una matriu

Actors: Usuari propietari de una matriu

Curs típic d'esdeveniments:

1. L'usuari selecciona una matriu.
 2. El sistema mostra una vista per configurar les mostres de la matriu.
 3. L'usuari selecciona una mostra i pot:
 - donar una data
 - donar una nom a la mostra
 - donar un origen de la mostra
 4. L'usuari accepta la configuració.
 5. El sistema guarda la configuració de la mostra.
-

Matriu007 Validar una matriu

Actors: Usuari propietari de una matriu

Curs típic d'esdeveniments:

1. L'usuari selecciona un matriu.
 2. El sistema mostra una vista per validar les dades.
 3. L'usuari accepta una validació.
 4. El sistema mostra al usuari si conte alguna dada buida com algun valor de mostres, un origen d'una mostra buit o una data de mostra buida.
-

Training001 Llistar els entrenaments del sistema

Actors: Usuari administrador

Curs típic d'esdeveniments:

1. L'usuari accedeix a la vista del llistat de entrenaments del sistema.
 2. El sistema llista els entrenaments amb les dades bàsiques:
 - nom de la matriu entrenada
 - estat del entrenament
 - descripció de l'entrenament
 - creador de l'entrenament
 - data de creació de l'entrenament.
-

Training002 Llistar els meus entrenaments

Actors: Usuari registrat

Curs típic d'esdeveniments:

1. L'usuari accedeix a la vista del llistat de entrenaments que ha creat.
2. El sistema llista els entrenaments que ha creat amb dades bàsiques:
 - nom de la matriu entrenada
 - estat de l'entrenament
 - descripció de l'entrenament
 - data de creació de l'entrenament.

Training003 Llistar matrius entrenables

Actors: Usuari registrat

Curs típic d'esdeveniments:

1. L'usuari accedeix a la vista del llistat de matrius entrenables.
2. El sistema llista els entrenaments amb un estat finalitzat i sense errors amb dades bàsiques:
 - nom de la matriu entrenada
 - estat de l'entrenament
 - creador
 - descripció de l'entrenament
 - data de creació.

Training004 Crear un entrenament

Actors: Usuari registrat

Curs típic d'esdeveniments:

1. L'usuari selecciona una matriu per entrenar.
2. El sistema li mostra un formulari per crear entrenaments.
3. L'usuari pot donar un nom, una descripció, seleccionar un origen dels disponibles i quines columnes vol entrenar. Finalment confirma les dades.
4. El sistema guarda l'entrenament i envia les dades al sistema de cues d'execucions de entrenaments. El sistema avalua si ha pogut enviar l'entrenament al sistema de cues en cas que el servei estigui caigut.
5. L'usuari veu les dades bàsiques de l'entrenament.

Training005 Re-enviar un entrenament al sistema de cues

Actors: Usuari propietari d'un entrenament

Curs típic d'esdeveniments:

1. L'usuari selecciona un entrenament que ha tingut problemes de enviament.
2. El sistema mostra una vista de visualització del entrenament.
3. L'usuari pot consultar quin possible error ha passat i pot confirmar el re-enviament.
4. El sistema actualitza les dades i re-envia les dades al sistema de cues.

Training006 Visualitzar un entrenament

Actors: Usuari registrat

Curs típic d'esdeveniments:

1. L'usuari selecciona d'un llistat un entrenament.
2. El sistema mostra una vista de visualització de l'entrenament amb el nom, descripció, data de creació i errors o resultats segons el cas.

Training007 Esborrar un entrenament

Actors: Usuari superadministrador, usuari propietari d'un entrenament

Curs típic d'esdeveniments:

1. L'usuari selecciona un entrenament per esborrar.
2. El sistema demana confirmació per esborrar l'entrenament.
3. L'usuari confirma l'acció
4. El sistema esborra el entrenaments i totes les prediccions que s'han fet a partir d'aquest entrenament.

Cursos alternatius:

- 3 L'usuari cancel·la la acció.
-

Training008 Descarregar els resultats d'un entrenament

Actors: Usuari registrat

Curs típic d'esdeveniments:

1. L'usuari selecciona un entrenament finalitzat.
 2. El sistema visualitza un enllaç amb la possibilitat de descarregar el fitxer resultats d'un entrenament.
 3. L'usuari accedeix a la descarrega.
 4. El sistema envia a l'usuari els resultats.
-

Training009 Actualitzar l'estat d'un entrenament

Actors: Cua

Curs típic d'esdeveniments:

1. La cua avisa al consumidor que ha finalitzat un entrenament i li envia les dades al consumidor.
 2. El consumidor rep les dades i li envia al sistema
 3. El sistema les guarda i actualitza l'estat del entrenament.
-

Prediction001 Llistar prediccions del sistema

Actors: Usuari administrador

Curs típic d'esdeveniments:

1. L'usuari accedeix a la vista del llistat de prediccions
2. El sistema llista totes les prediccions amb dades bàsiques:
 - nom de la matriu
 - nom de l'entrenament
 - nom de la predicció
 - data de creació
 - estat de la execució.

Prediction002 Llistar les meves prediccions

Actors: Usuari registrats

Curs típic d'esdeveniments:

1. L'usuari accedeix a la vista del llistat de prediccions creades per ell.
2. EL sistema llista totes les prediccions creades per l'usuari amb dades bàsiques.
 - nom de la matriu
 - nom de l'entrenament
 - nom de la predicció
 - data de creació
 - estat de la execució.

Prediction003 Esborrar una predicció

Actors: Usuari registrats

Curs típic d'esdeveniments:

1. L'usuari selecciona una predicció per esborrar.
2. El sistema li demana confirmació per esborrar la predicció.
3. L'usuari confirma la acció.
4. El sistema esborra la predicció.

Cursos alternatius:

- 3 L'usuari cancel·la la acció.
-

Prediction004 Crear una matriu de predicció des d'un fitxer

Actors: Usuari registrats

Curs típic d'esdeveniments:

1. L'usuari selecciona un entrenament per crear una predicció.
2. El sistema li mostra un formulari de creació de prediccions.
3. L'usuari accedeix a un formulari on pot donar nom a la matriu de predicció i seleccionar una fulla de càlcul. L'usuari accepta el formulari.
4. El sistema crea la matriu on:
 - La primera fila del fitxer s'associa com una variable entrenada. Si la variable es igual al identificador de la variable entrenada, automàticament s'assigna a aquesta columna a aquesta columna de l'entrenament i al un conjunt de fitxers d'envelliments que s'ha entrenat. Les dues ultimes columnes son optatives, on s'especifica l'origen i la data de la mostra.
 - Cada fila del fitxer, després de la primera fila:
 - La primera columna és l'identificador de la mostra. Si conté un alias d'origen, automàticament s'assigna un origen
 - Les columnes restants s'assignen com a valors de la mostra
 - Les dos ultimes columnes seran l'origen o la data segons la primera fila del fitxer.
5. El sistema mostra la matriu de predicció.

Prediction005 Actualitzar una matriu de predicció des d'un fitxer

Actors: Usuari propietari de la predicció

Curs típic d'esdeveniments:

1. L'usuari selecciona un entrenament.
2. El sistema li mostra un formulari d'edició de prediccions.
3. L'usuari pot actualitzar el nom de la predicció i seleccionar una fulla de càlcul i guardar els canvis.
4. El sistema esborra la matriu de predicció, actualitza les dades i crea la matriu on:
 - La primera fila del fitxer s'associa com una variable entrenada. Si la variable es igual al identificador de la variable entrenada, automàticament s'assigna a aquesta columna a aquesta columna de l'entrenament i al un conjunt de fitxers d'envelliments que s'ha entrenat. Les dues ultimes columnes son optatives, on s'especifica l'origen i la data de la mostra.
 - Cada fila del fitxer, després de la primera fila:
 - La primera columna és l'identificador de la mostra. Si conté un alias d'origen, automàticament s'assigna un origen
 - Les columnes restants s'assignen com a valors de la mostra
 - Les dos ultimes columnes seran l'origen o la data segons la primera fila del fitxer.

Configurar una mostra d'una matriu de predicció

Prediction 006 Configurar una mostra d'una matriu de predicció

Actors: Usuari propietari de la predicció

Curs típic d'esdeveniments:

1. L'usuari selecciona una predicció i una mostra.
 2. El sistema mostra una vista de edició de la matriu.
 3. L'usuari pot donar un data, un nom i un origen a la mostra i guardar els canvis.
 4. El sistema guarda les dades.
-

Enviar una predicció al sistema de cues

Prediction007 Enviar una predicció al sistema de cues

Actors: Usuari propietari de la predicció

Curs típic d'esdeveniments:

1. L'usuari selecciona una predicció.
 2. El sistema li mostra un formulari per enviar la matriu de predicció al sistema de cues.
 3. L'usuari confirma la acció.
 4. El sistema prepara les dades per enviar i les envia al sistema de cues d'execucions de prediccions.
-

Veure una predicció

Prediction008 Veure una predicció

Actors: Usuari registrat

Curs típic d'esdeveniments:

1. L'usuari selecciona una predicció.
2. El sistema li mostra les dades bàsiques de la predicció i la matriu de predicció.

Veure els resultats d'una predicció

Prediction009 Veure els resultats d'una predicció

Actors: Usuari registrat

Curs típic d'esdeveniments:

1. L'usuari selecciona els resultats d'una predicció.
 2. El sistema li mostra les diferents sortides de la execució.
-

Actualitzar l'estat d'una predicció

Prediction010 Actualitzar l'estat d'una predicció

Actors: Cua, Consumidor

Curs típic d'esdeveniments:

1. La cua avisa al consumidor que ha finalitzat una predicció i li envia les dades al consumidor
 2. El consumidor rep les dades i li envia al sistema
 3. El sistema les guarda i actualitza l'estat de la predicció.
-

Configurar una columna d'una matriu de predicció

Prediction011 Actualitzar l'estat d'una predicció

Actors: Propietari d'una predicció

Curs típic d'esdeveniments:

1. L'usuari selecciona una matriu de predicció.
 2. El sistema mostra una vista per configurar les columnes de una matriu de predicció.
 3. L'usuari selecciona una columna i pot:
 - donar un nom a la columna
 - seleccionar una variable amb el conjunt d'envelliments de la variable que s'ha entrenat.
 4. L'usuari accepta la configuració
 5. El sistema salva la configuració de la columna i notifica a l'usuari.
-

3.3 Model Conceptual

A continuació es descriu el model conceptual resultant dels casos d'usos.

3.3.1 Diagrama de classes

La figura 3.3 és el diagrama UML on està implementat el modelatge de dades.

3.3.2 Explicació de les classes

A continuació es descriuran algunes de les classes i alguns atributs per a la seva millor comprensió.

- *Users*: en aquesta classe es guarda la informació dels usuaris.
- *Groups*: en aquesta classe es guarda la informació dels grups.
- *Season*: en aquesta classe es guarda la informació d'un fitxer d'envelliment.

- *SeasonSet*: en aquesta classe es guarda la informació d'un conjunt de fitxers.
- *SeasonSetComponent*: aquesta classe associa fitxers als conjunts de fitxers.
- *Matrix*: en aquesta classe es guarda la informació bàsica d'una matriu.
- *Sample*: en aquesta classe es guarda la informació de cadascuna de les files de la matriu a la que pertany mitjançant la associació *rows*.
- *VariableMatrixConfig*: aquesta classe guarda la configuració d'una columna de la matriu i associa la columna a una variable i a un conjunt de fitxers d'aquesta variable.
- *Training*: en aquesta classe es guarda la informació d'un entrenament.
 - requestId: identificador del procés en la cua d'execució.
 - status: estat de la predicció. Mirar l'apartat 3.4.
- *Column Selected*: aquesta classe associa un entrenament amb quines columnes es volen entrenar.
- *PredictionMatrix*: en aquesta classe es guarda la informació d'una matriu de predicció.
 - requestId: identificador del procés en la cua d'execució.
 - predictionResult: collecció de resultats de la execució d'una predicció.
 - status: estat de la predicció. Mirar 3.4.
- *PredictionSample*: en aquesta classe es guarda la informació de les mostres(files) d'una matriu de predicció.
- *PredictionColumn*: en aquesta classe es guarda la informació de les columnes d'una matriu de predicció i la associació entre les columnes entrenades. La propietat index conté la posició que ocupa en la matriu.

3.4 Model d'estats

La aplicació i el sistema de cues de execucions son dos components separats. Aixó ens obliga a definir estats per els entrenaments i per les prediccions.

El principal problema que existeix en la versió actual de Ichnaea Software és que no retorna estats parcials d'execucions: no ens retorna si ha començat la execució o quan temps queda o percentatge porta. A continuació s'explica quins estats s'han definit per els entrenaments i per les prediccions.

3.4.1 Estats dels entrenaments

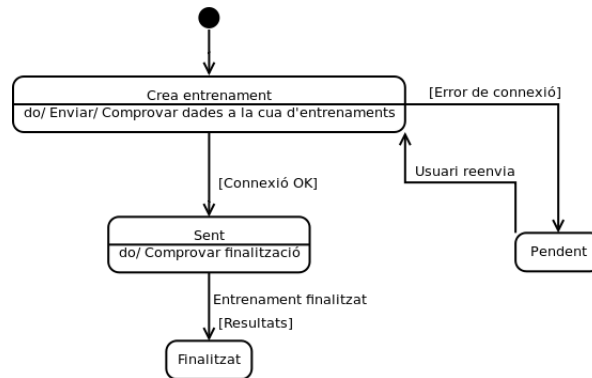


Figura 3.4: Diagram d'estats dels entrenaments

Per crear un entrenament, primer es comprova que es pot establir connexió amb la cua. Si no és pot, l'entrenament queda marcat amb l'estat pendent. En el cas que es pugui establir connexió, s'envien les dades i es queda en aquest estat fins que li arribi algun esdeveniment de finalització amb els resultats o amb un error d'Ichnaea.

En el cas que estigui pendent, l'usuari podrà re-intentar enviar l'entrenament a la cua després de comprovar l'error.

3.4.2 Estats de les prediccions

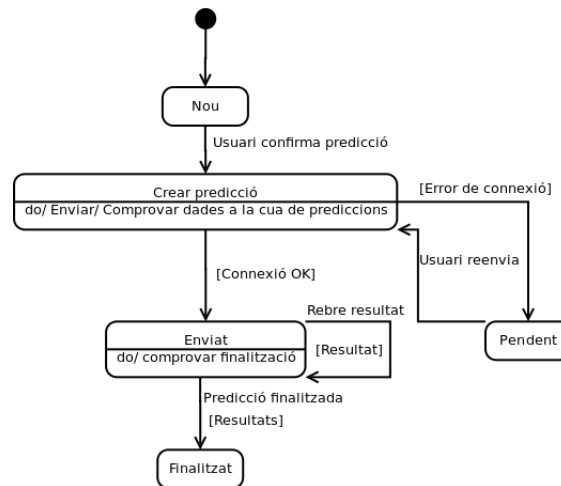


Figura 3.5: Diagram d'estats dels entrenaments

Per crear una predicció, primer s'ha de configurar la matriu. Mentre l'usuari estigui configurant la matriu de predicció estarà en un estat inicial(*nou*). Quan l'usuari confirmi l'enviament, primer es comprova que es pot establir connexió amb la cua d'execucions de prediccions. Si no és pot, l'entrenament queda marcat amb l'estat pendent. En el cas que es pugui establir connexió, s'envien les dades. Es queda en aquest estat rebent múltiples respostes fins que li arribi algun esdeveniment de finalització amb els resultats o amb els errors.

En el cas que estigui pendent, l'usuari podrà re-intentar enviar la predicció a la cua després de comprovar l'error.

3.5 Model del comportament

En aquesta secció donarem veurem els diagrames de seqüència de les operacions més complexes dels casos d'us.

Crear un conjunt de fitxers d'envelliments per una variable

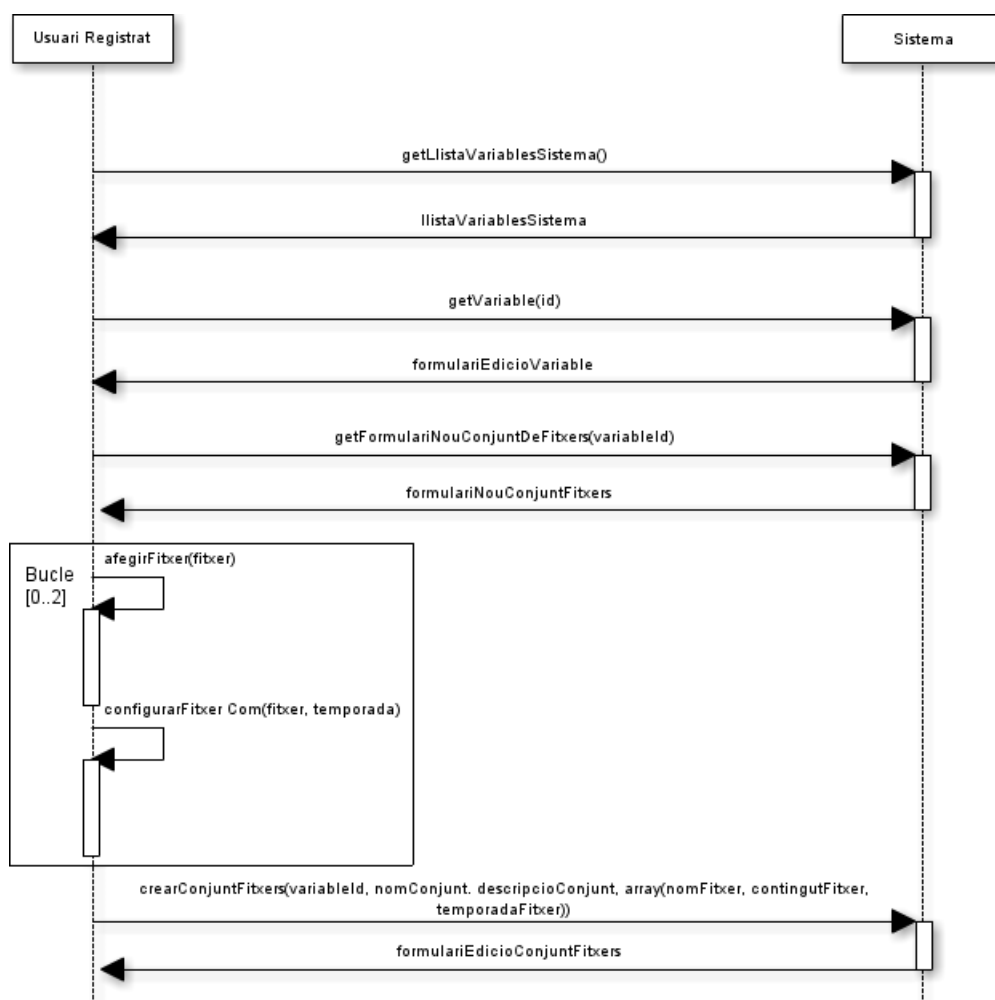


Figura 3.6: Diagram de seqüència *Crear un conjunt de fitxers d'envelliments per una variable*

Afegir un nou fitxer a un conjunt de fitxers d'envelliments

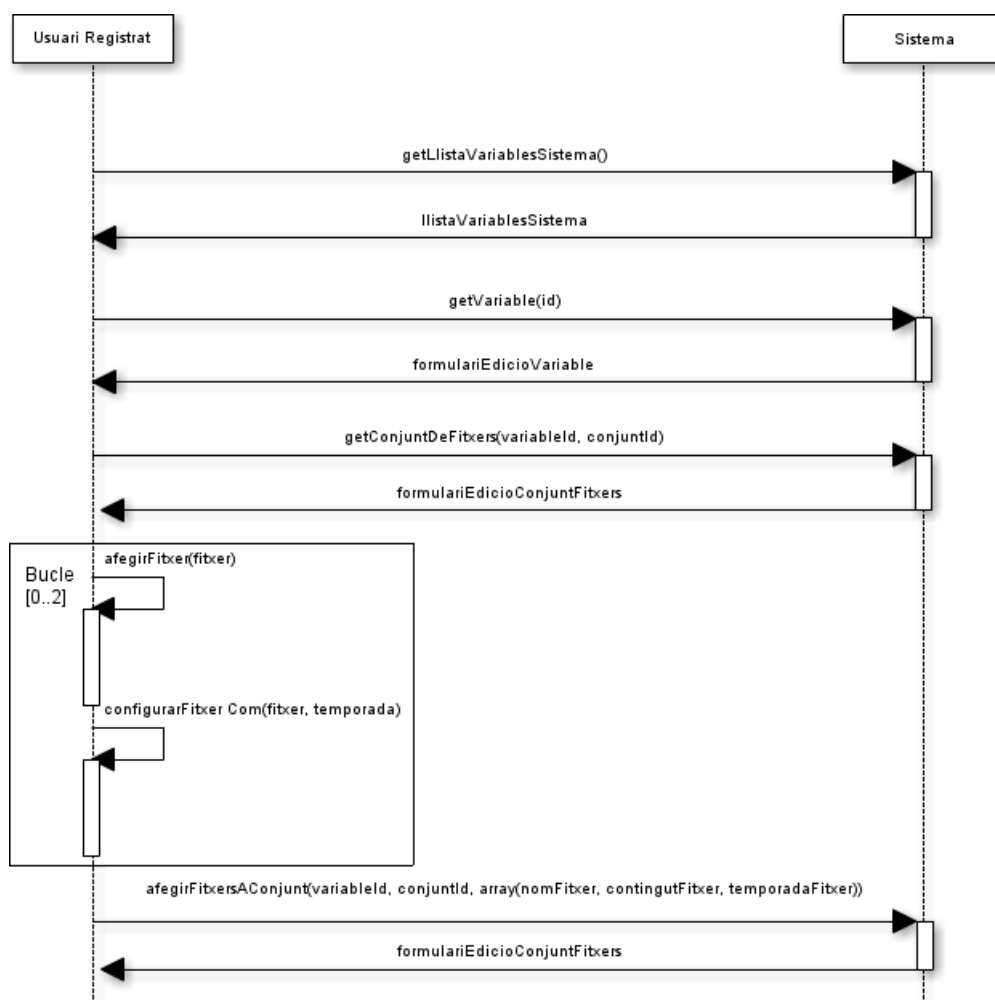


Figura 3.7: Diagram de seqüència *Afegir un nou fitxer a un conjunt de fitxers d'envelliments*

Crear una matriu des d'un fitxer

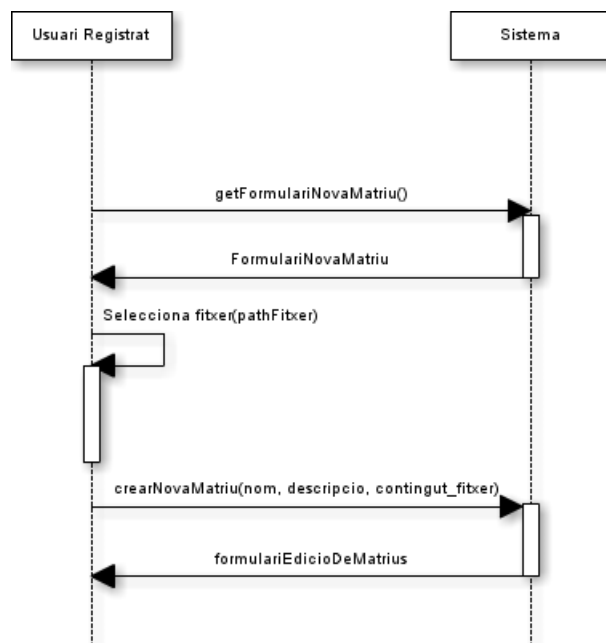


Figura 3.8: Diagram de seqüència *Crear una matriu des d'un fitxer*

Crear un entrenament

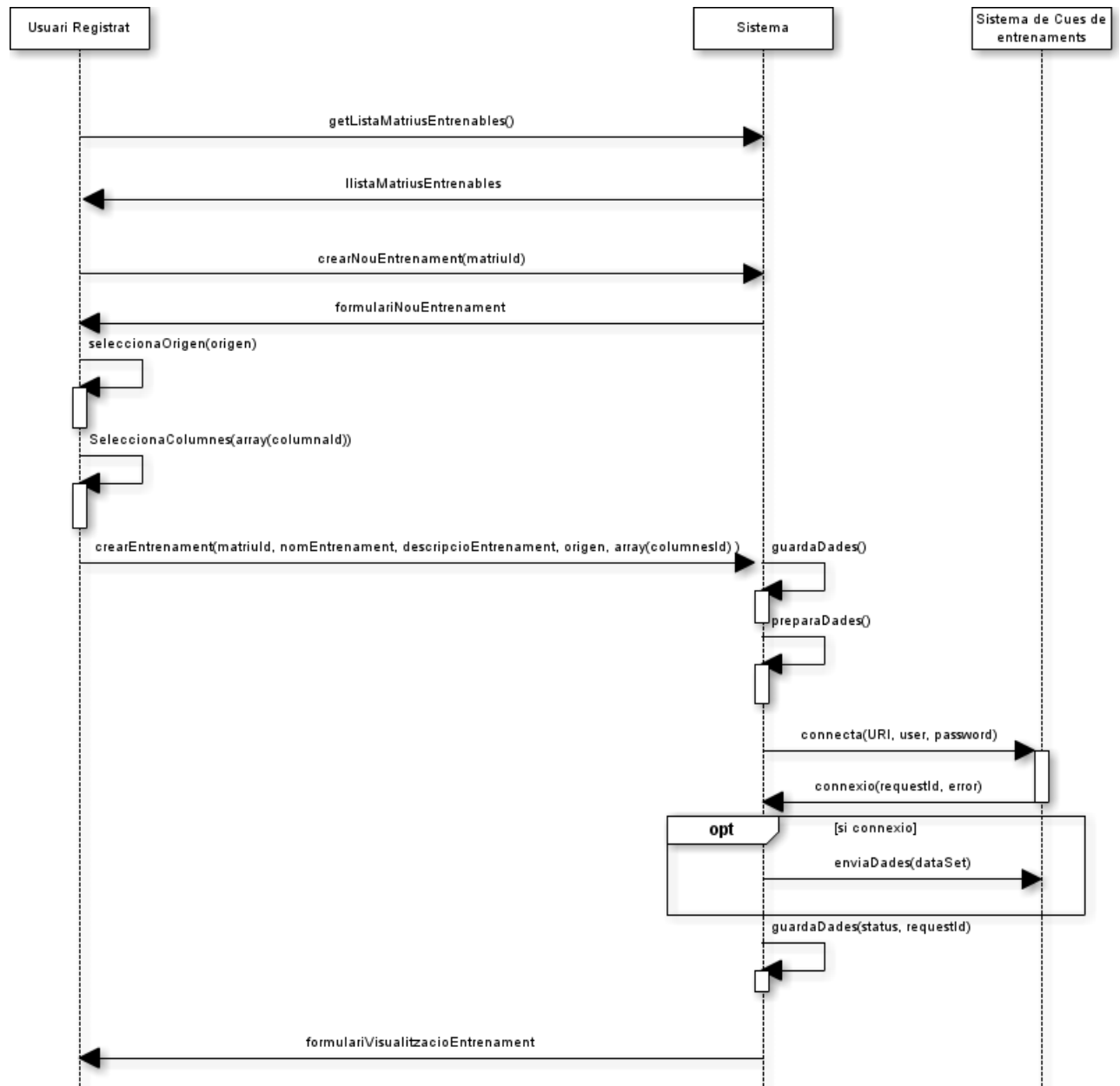


Figura 3.9: Diagram de seqüència *Crear un entrenament*

Configurar una columna d'una matriu

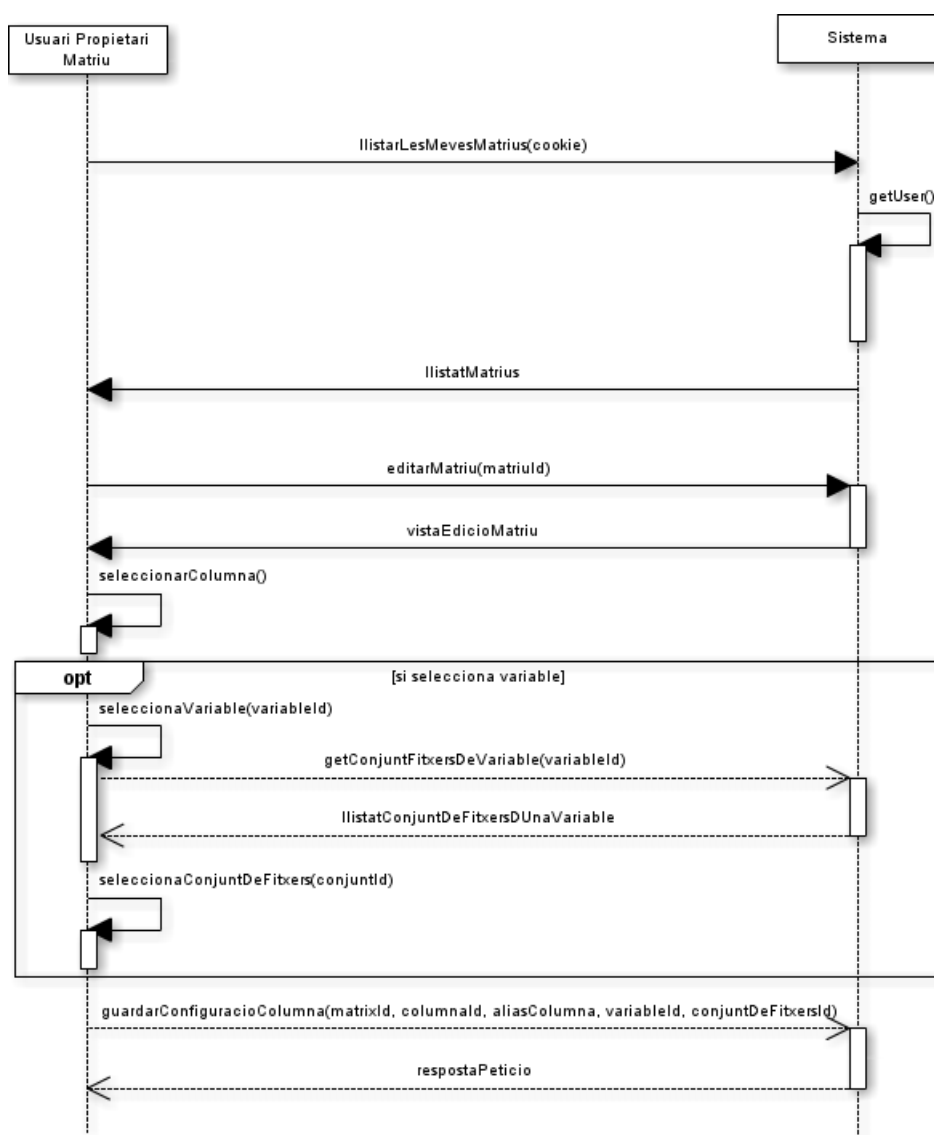


Figura 3.10: Diagrama de seqüència *Configurar una columna d'una matriu*

Actualitzar l'estat d'una predicció

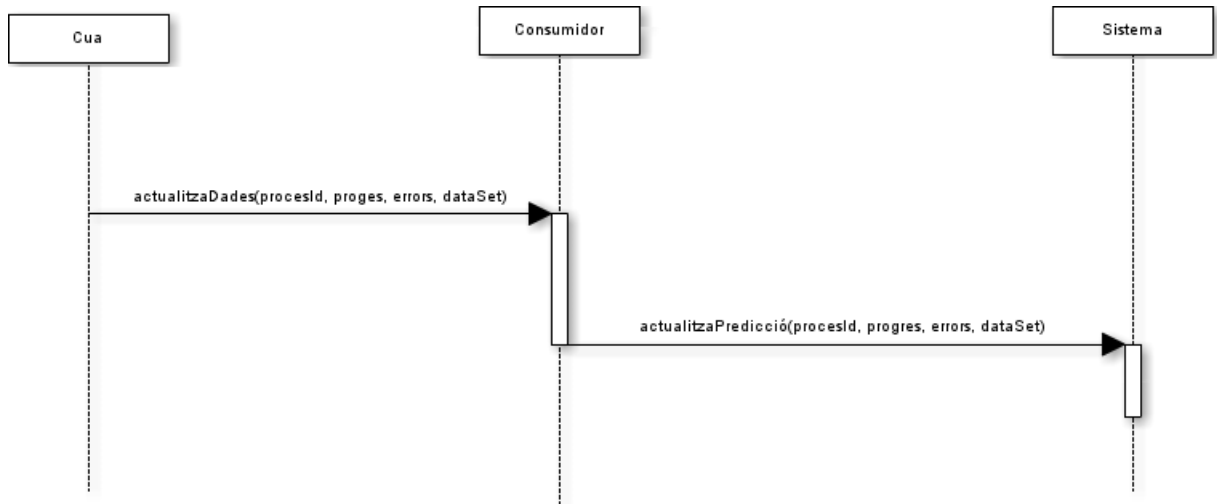


Figura 3.11: Diagrama de seqüència *Actualitzar l'estat d'una predicció*



Figura 3.2: Diagrama de casos d'ús

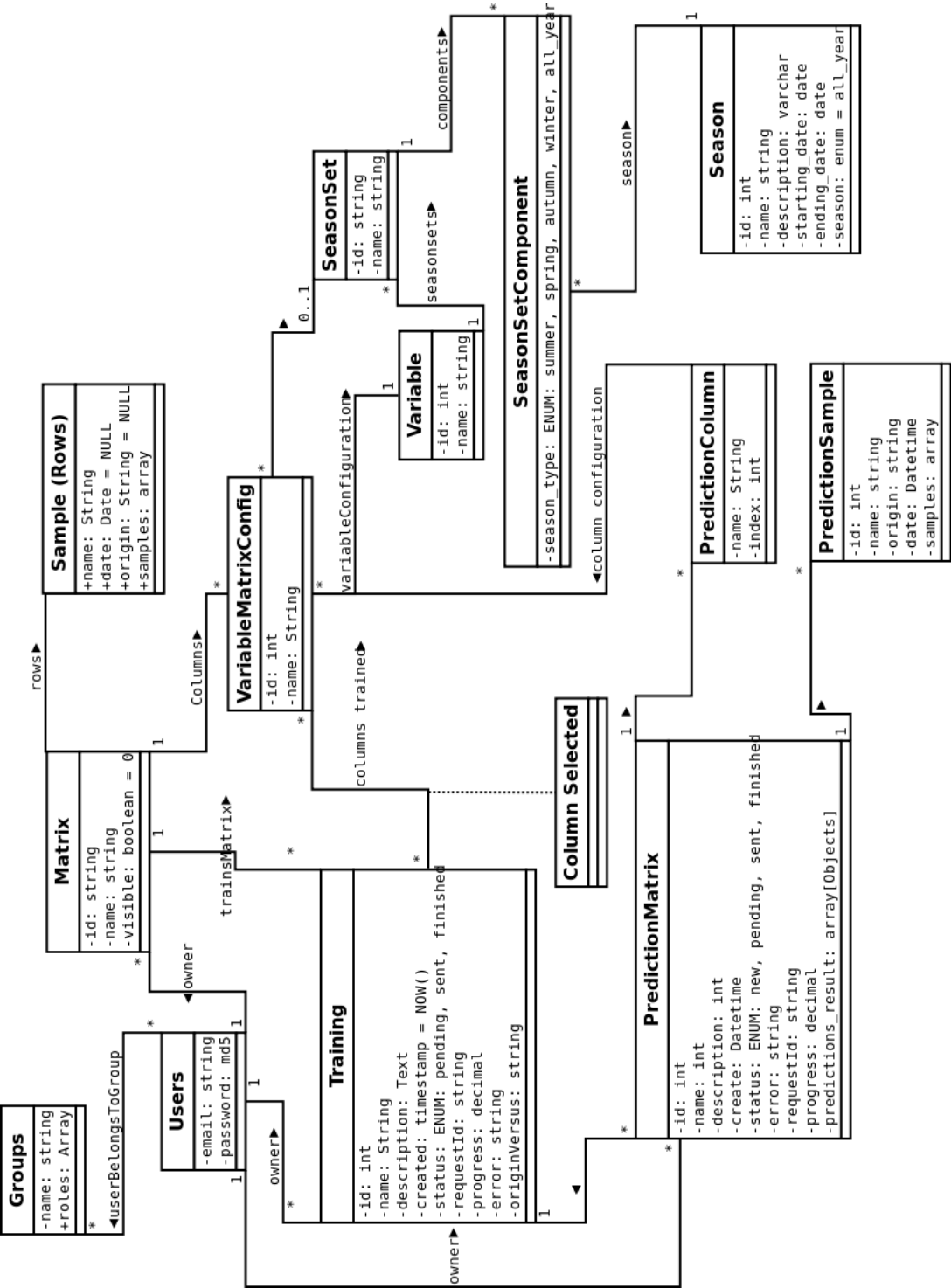


Figura 3.3: Model de dades

Capítol 4

Disseny

En aquest capítol veurem els patrons de dissenys emprats i les tecnologies usades. També descriurem les interfícies d'usuari més complexa.

4.1 Esquema general lògic arquitectònic del sistema

La figura 4.1 és un esquema dels components lògics del sistema on:

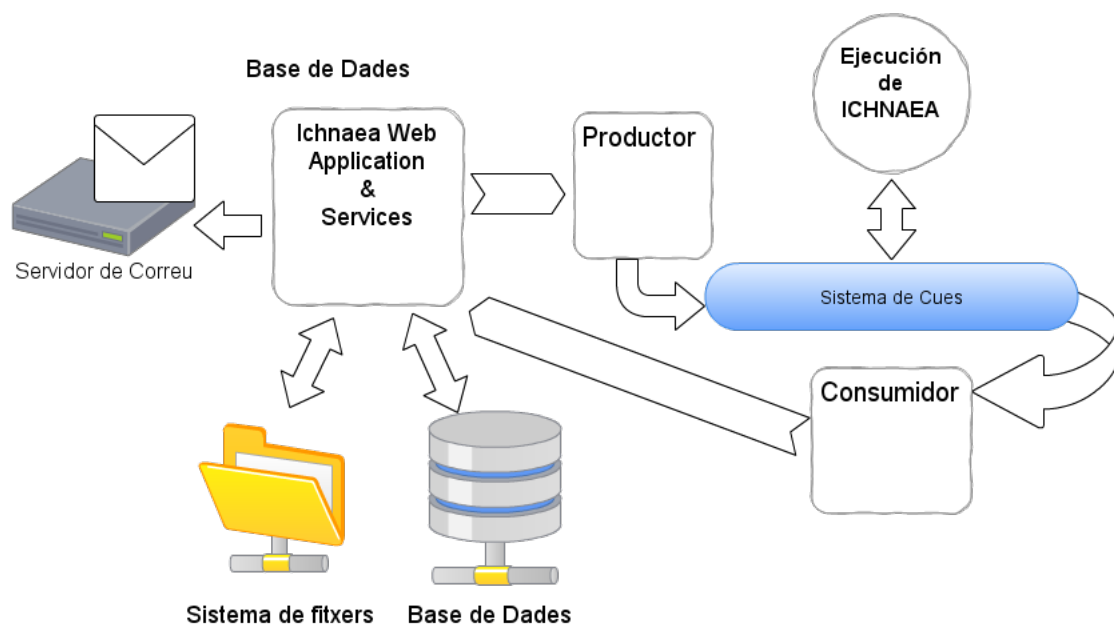


Figura 4.1: Arquitectura del sistema

- *Ichnaea Web Application & Services*: és el *core* de la aplicació i dels serveis HTTP.
- Servidor de correu és el servidor SMTP per enviar correus electrònics als usuaris.
- Productor-Cua-Consumidor és el sistema de cues. Expliquem el paradigma al capítol 5.4.1. La funció és gestionar els inicis, fluxos i finals d'execucions de Ichnaea.
- La base de dades és on es guarden els continguts i models de dades i resultats.
- El sistema de fitxers és on es guarden els resultats dels entrenaments per usar amb les prediccions.

4.2 Patró de disseny

Per la implementació del sistema web s'han usat un disseny per capes (Presentació, domini i persistència de dades) amb els següents patrons de disseny:

- Model-Vista-Controlador amb controlador frontal
- Capa de Servei
- Injecció de dependències
- Repositori de model de dades
- Capa de mapejat de dades
- View template
- Interfícies enriquides amb servei webs

4.2.1 Esquema del disseny

La figura 5.1 és el diagrama general dels patrons que s'usaran per aquest projecte.

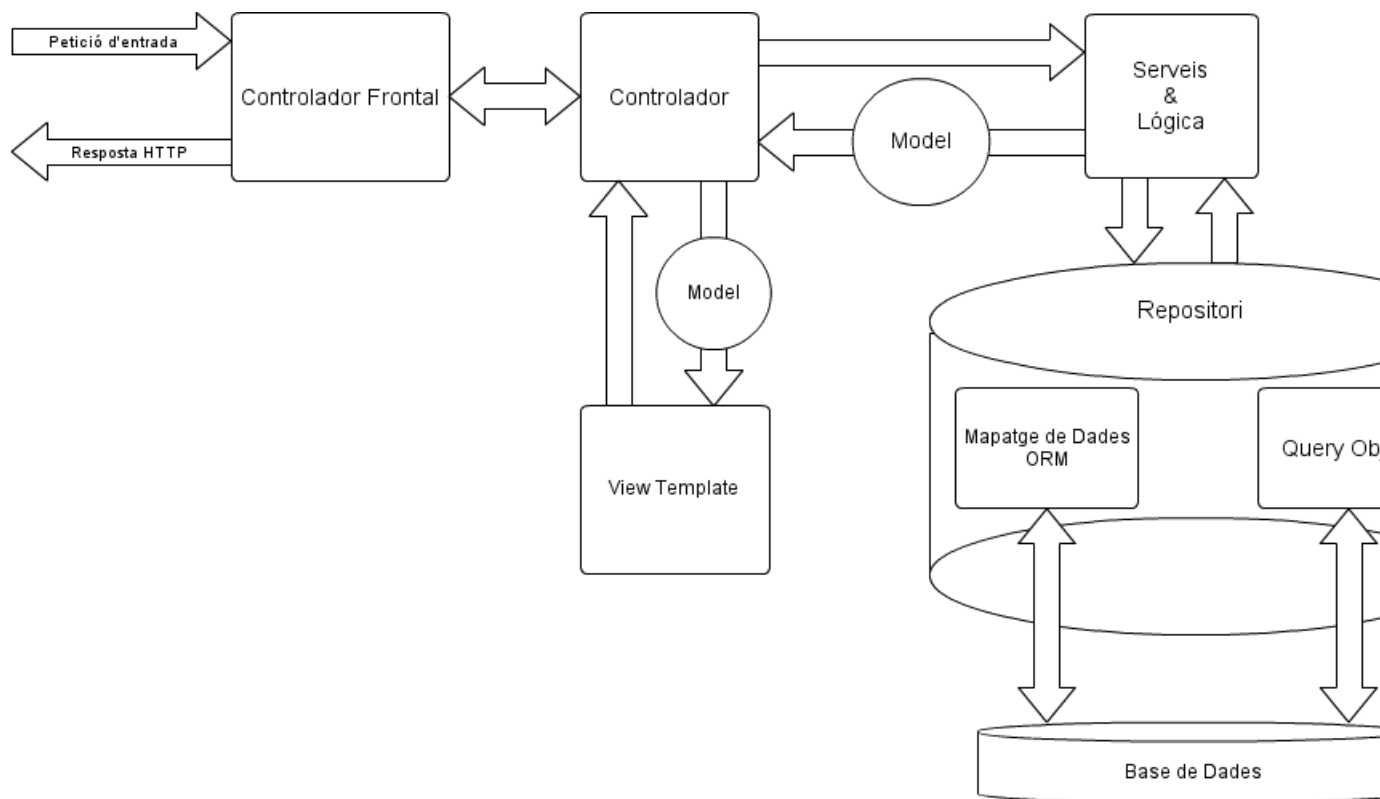


Figura 4.2: Patrons de disseny

4.2.2 Explicació del disseny

El patró MVC, Model-Vista-Controlador, amb controlador frontal és un disseny típic en les aplicacions web on:

- el **model** és una representació de les dades.
- la **vista** transforma el model a un format visible i llegible.
- el **controlador frontal** rep totes les peticions i les re-dirigeix als controladors corresponents
- el controlador se encarrega de processar les peticions específiques

L'arquitectura **MVC** separa la capa de presentació de la lògica de domini. La capa de presentació accedeix a la capa de domini mitjançant serveis, injectant dependències.

La **injecció de dependències** és un patró a on es subministren objectes a una classe en lloc de ser la classe qui crea els objectes.[14] Les avantatges de usar DI("dependency injection"):

- Codi més fàcil de mantenir, estendre o modificar
- Desenvolupament guiat per proves (Test Driven Development o TDD en anglés)
- DI ens obliga a planejar una mica millor les nostres dependències; decidim si una classe realment necessita d'un altre objecte per realitzar la seva funció.

En la **capa de serveis** conté la lògica de Ichnaea. Accedeixen a les dades mitjançant **repositoris** d'objectes. Els **repositoris** son mediadors entre el domini i les dades persistents. Els repositoris retornen entitats i/o col·leccions d'entitats de la lògica de domini.

El **mapatge de dades** ORM("Object Relational Mapping"), mapatge objecte-relacional, és un patró de disseny(encara que alguns enginyers els agrada dir que és una tècnica de programació i a uns altres una tecnologia) que estableix una relació directe entre les entitats i la dades persistents.[13]

4.3 Disseny d'interfícies

En el disseny de les interfícies, es pretén definir com serà la part del sistema que interactua amb l'usuari. Consisteix en definir els mecanismes amb els quals els usuaris podran interactuar amb el sistema i la manera de mostrar la informació a l'usuari.

4.3.1 Disseny principal

A continuació es detalla la estructura estètica de la aplicació. **DIAGRAMA ESQUEMA DEL MENU** El menú serà condicional als permisos de l'usuari que estigui usant. La jerarquia del menu es la següent:

- Logo
- Matrius
 - Les meves matrius
 - Crear una matriu
- Entrenaments
 - Els meus entrenaments

- Crear un entrenament
- Prediccions
 - Les meves prediccions
 - Crear una predicció
- Dades de Ichnaea
 - Variables del sistema
 - Fitxers
 - Matrius
 - Llista de entrenaments predictibles
- Administració: Solament disponible per usuaris administradors
 - Usuaris
 - Crear una variable
 - Eines administratives de control de cues
 - Llista d'entrenaments
 - Llista de prediccions
- Usuari
 - Casa de l'usuari
 - Canvi de contrasenya
 - Desconnectar-se

4.3.2 Interfície de configuració de matrius

El disseny d'interfícies per les configuracions de les matrius es basa en funcionalitats enriquides[18]. Per dotar de funcionalitats a les interfícies webs s'utilitza peticions asíncrones.

Per complir aquest requeriment s'ha de dissenyar una llibreria per atendre les peticions asíncrones i desenvolupar funcionalitats per atendre les respostes a aquestes peticions. A continuació es descriuen les interfícies més complexes: les responsables de la configuració de les matrius.

Interfície de configuració de matrius

Les funcionalitats enriquides que han de complir son:

- Carrega de conjunt de fitxers segons les variables.
- Guardar la configuració de una columna: alias, variable i conjunt de fitxers.
- Desplegar un calendari per seleccionar una data i assignar-la a una mostra.
- Desplegar un selector obert de possibles valors de tots els orígens que estiguin especificats a les mostres.
- Actualitzar el valor d'una mostra i d'una variable.
- Permetre la mobilitat per la matriu fixant les capçaleres de les columnes i les files.
- Adaptar-se a les dimensions de la pantalla.

A la figura 4.3 s'especifica el disseny de la interfície de configuració de matrius.

Configuració

Matriu

Acció 1	Acció 2	Acció n
---------	---------	---------

Missatges del sistema

Conjunt de fixers ▾

Conjunt de fixers ▾

Ichnaea Variables ▾

Ichnaea Variables ▾

Variable Alias

OK

Variable Alias

OK

<div style="display: flex; justify-content: space-between;"> <div>Mostra ID</div> <div>Calendari</div> <div>Origens ▾</div> </div>	Sample 4	Sample editable
<div style="display: flex; justify-content: space-between;"> <div>Mostra ID</div> <div>Calendari</div> <div>Origens ▾</div> </div>	Sample 5	Sample 5
<div style="display: flex; justify-content: space-between;"> <div>Mostra ID</div> <div>Calendari</div> <div>Origens ▾</div> </div>	Sample 6	Sample 6

Ajustable segons zoom i tamany de la pantalla

Ajustable segons zoom i tamany de la pantalla

Figura 4.3: Interfície de configuració de matrius

Interfície de configuració de matrius de predicció

Les funcionalitats enriquides que han de complir:

- Guardar la configuració de una columna: alias i assignació d'una columna entrenada.
- Desplegar un calendari per seleccionar una data i assignar-la a una mostra.
- Desplegar un selector obert de possibles valors de tots els orígens que estiguin especificats a les mostres.
- Actualitzar el valor d'una mostra i d'una variable.
- Adaptar-se a les dimensions de la pantalla.

A la figura 4.4 s'especifica el disseny de la interfície de configuració de matrius de prediccions.



Figura 4.4: Interfície de configuració de matrius de prediccions

Capítol 5

Implementació

En aquest capítol es descriuen les tecnologies estudiades i aplicades per implementar el disseny anteriorment descrit.

5.1 Estudi previ de les tecnologies

En aquesta secció es farà una descripció de les tecnologies estudiades per tal de poder realitzar aquest projecte.

5.1.1 *Framework* a mida

En un principi es va valorar la possibilitat de desenvolupar tot un nou *framework* per la implementació del projecte. Les tecnologies emprades són PHP com a llenguatge de programació i MySQL com a motor de bases de dades.

Avantatges

- La principal avantatge és el control total de la implantació de tots els processos i tecnologies.

Desavantatges

- Desenvolupament extremadament lent.
- Re-inventar la roda quan no és necessari.

5.1.2 *Codeigniter*

Codeigniter és un *framework* per a desenvolupament web ràpid amb PHP.[19] Usa el patró "active record" per la gestió d'emmagatzematge de dades.[20]

Avantatges

- Corba d'aprenentatge ràpida.
- Desenvolupament ràpid.
- És patró MVC.

Desavantatges

- No compleix tots el requisits de disseny de patrons especificat.
- No inclou *templating*
- Dificultat en segregar la lògica de la presentació.

5.1.3 *Symfony2*

Symfony2 és un HTTP *framework* per a PHP. Nativament implementa una variació del Model-Vista-Controlador amb controlador frontal amb injecció de dependències a la capa de serveis.[3]

Avantatges

- Altament configurable.
- Compleix tots els requisits de dissenys de patrons especificat.
- Alt rendiment

Desavantatges

- Corba d'aprenentatge alta.

5.2 Implementació

El framework escollit ha sigut *Symfony2*. A continuació es dona una visió general de les tecnologies.

5.2.1 *Symfony2*

Arquitectonicament, *Symfony2* estructura el codi en *Bundles*, similar als paquets de JAVA. Els *bundles* son un conjunt de serveis, entitats i recursos independents entre si. Els *bundles* implementats son:

- *Bundle* de usuaris: UserBundle. Paquet de serveis, vistes i recursos per usuaris. S'ha utilitzat un paquet de *Symfony2*: FOSUserBundle [22]. S'ha fet una extensió del paquet per implementar la gestió de rols i grups.
- *Bundle* de matrius: MatrixBundle. Paquet de serveis, vistes i recursos per les matrius.
- *Bundle* de trainings: TrainingBundle. Paquet de serveis, vistes i recursos per els entrenaments.
- *Bundle* de serveis webs: ApiBundle. Paquet de serveis, vistes i recursos per la API JSON Restful.
- *Bundle* de predicció: PredictionBundle. Paquet de serveis, vistes i recursos per les matrius de prediccions.

5.2.2 Gestió de dependències

Symfony2 utilitza *Composer* per la gestió de dependències.[21] *Composer* és un gestor de dependències a nivell d'aplicació. Les dependències son:

- JQuery: llibreria Javascript que s'executa en el costat client per enriquir les interfícies. Simplifica la manera de interaccionar amb els documents HTML i la simplifica la manera de manipular l'arbre DOM.[5]
- FOSUserBundle: paquet per a *Symfony* per la gestió de usuaris.[22]
- Bootstrap: framework pel *front-end* desenvolupat per Twitter. Implementa nous estàndards com HTML5 i CSS3. Permet el desenvolupament d'interfícies de manera ràpida i usables suportades per múltiples navegadors.
- RestBundle: paquet per *Symfony* per desenvolupar llibreries API JSON Restful.
-
- Symfony FS: paquet per gestionar sistema de fitxers.

- Doctrine Fixtures: paquet per gestionar la inserció controlada de dades a la base de dades.
- TWIG: motor de plantilles per a PHP.[8]
- Doctrine: ORM pel mapatge de dades, usant MySQL.[7]

5.2.3 Recursos

La estructura de recursos de la aplicació és la següent:

Matrius	matrix/(id)
Entrenamens	matrix/(id)/training/(id)
Prediccions	matrix/(id)/training/(id)/prediction/(id)

S'ha emprat aquesta estructura de recursos degut a les dependències entre les diferents entitats. Un entrenament depén d'una matriu i una predicció depén d'un entrenament.

5.3 API: llibreria de serveis web

S'ha desenvolupat una llibreria API JSON RESTful per donar resposta a les funcionalitats asíncrones de les interfícies.[4] S'ha emprat aquesta tecnologia per la escal.labilitat que aporta i perquè en un futur es pugui aprofitar el desenvolupament d'aquesta amb la integració d'altres tipus client, com poden ser aplicacions mòbils o altres serveis.

La arquitectura RESTful aprofita al màxim el protocol HTTP per establir dades i actualment s'usen per donar funcionalitats asíncrones a les interfícies(mirar 4.3).

Les operacions, els recursos i el paràmetres implementats son:

OPERACIÓ	RECURS	ENTRADA	SORTIDA	HTTP CODIS
GET	/api/season/{id}		{'season': SeasonObject}	200, 401
POST	/api/season/searchByName	{pattern: string}	[string-match-1, ... string-match-n]	200, 401
GET	/api/variable/{id}/seasonSet		{'seasonSet': SeasonSetObject}	200, 401
DELETE	/api/variable/{id}/seasonSet/{id}		{'status': int, 'message': string}	200, 401, 404
DELETE	/api/variable/{id}/seasonSet/{id}/component/{id}		{'status': int, 'message': string}	200, 401, 404
DELETE	/api/variable/{id}/seasonSet/{id}/component/{id}/complete		{'status': int, 'message': string}	200, 401, 404
PUT	/api/matrix/{id}/column/{id}		{'status': int, 'message': string}	200, 401
PUT	/api/matrix/{id}/sample/{id}		{'status': int, 'message': string}	200, 401
PUT	/api/matrix/{id}/column/{id}	{name: string, variable: int, season: int}	{'status': int, 'message': string}	200, 401
PUT	/api/matrix/{id}/sample/{id}	{name: string, 'date': Date, 'origin': string}	{'status': int, 'message': string}	200, 401
PUT	/api/matrix/{matrix_id}/sample/{sample_id}/index/{index}	{data: string}	{'status': int, 'message': string}	200, 401
PUT	/api/prediction/{id}/sample/{id}	{name: string, variable: int}	{'status': int, 'message': string}	200, 401
PUT	/api/prediction/{id}/sample/{id}/index/{index}	{data: string}	{'status': int, 'message': string}	200, 401
PUT	/api/prediction/{id}/column/{index}	{name: string, 'variable': int}	{'status': int, 'message': string}	200, 401

Figura 5.1: Operacions de serveis web

On:

- Operació és el mètode HTTP.
- Recurs és el identificador del recurs.
- Entrada és l'objecte JSON d'entrada de la operació.
- Sortida és l'objecte JSON de sortida de la operació.
- HTTP Codi son els codis HTTP que retornen.[23]

Totes les operacions estan securitzades per *cookies*[24] i per sessió per identificar i validar a l'usuari.[25]

5.4 Integració amb el sistema de cues RabbitMQP

5.4.1 Introducció a l'arquitectura de cues: AMQP

L'estàndard AMQP ("Advanced Message Queuing Protocol") és un protocol d'estàndard obert en la capa d'aplicacions d'un sistema de comunicació. Les característiques que defineixen al protocol AMQP son la orientació a missatges, encuament("queuing"), enrutament i exactitud, entre altres com, per exemple, la seguretat o les subscripcions.[11]

AMQP defineix una sèrie d'entitats. Des de la perspectiva de la interconnexió les més rellevants son:

- El corredor de missatges: un servidor on els clients AMQP es connecten usant el protocol AMQP. Els corredores de missatges poden executar-se en un entorn distribuït, però aquesta capacitat és específica de la implementació.
- Usuari: un usuari és una entitat amb credencial pot ser autoritzat a connectar-se a un corredor.
- Connexió: una connexió física, usant per exemple TCP/IP, entre el corredor i l'usuari.
- Clients: productors i consumidors. **EXPLICAR PRODUCTORS I CONSUMIDORS.**

RabbitMQ és un *software* que implementa aquest protocol. El sistema de cues està implementat per Miguel Ibero. I ofereix una llibreria per la seva integració.

5.4.2 Consumidors i gestió de resultats

Les respostes de les execucions de Ichnaea enviades per la cua (mirar la figura ??), les rep el consumidor.

Symfony2 permet crear comandes CLI per crear processos. Els consumidors necessiten ser processos "stand-alone"(autònoms) que escoltin les sortides, consumeixin, els resultats de la cua.

L'esquelet d'una comanda per aquests consumidors actualment és:

```
class ConsumerCommand extends ContainerAwareCommand
{
    //Definició de la comanda
    protected function configure()
    {
        $this
            ->setName('nom_de_la_comanda')
            ->setDescription('Consumer server');
    }

    //Execució de la comanda
    protected function execute(InputInterface $input,
        OutputInterface $output)
    {
        //Interfície per integrar el sistema de cues
        $amqp = new
            AmqpConnection('URI_per_fer_la_connexio');
        $amqp->open();
    }
}
```

```
        //Crida al servei
        $servei =
            $this->getContainer()->get('nom_de_servei');

        //Crear el consumidor, amb una
        //funcio que es crida quan la
        //cua estableix comunicacio amb
        //un objecte de la resposta esperada
        //i li passa el servei
        $amqp->listenForBuildModelResponse(function
            (ObjectResponse $resposta_de_la_cua) use
            ($servei){
                $servei->actualitzaDades($resposta_de_la_cua);
            });
        $amqp->wait();
    }
}
```

Consumidor d'entrenaments

El consumidor d'entrenaments es queda escoltant les respostes del servei de

Consumidor de prediccions

El consumidor de prediccions es queda escoltant les respostes del servei de

Capítol 6

Proves i tests

En aquest capítol exposem les proves realitzades de la aplicació.

6.1 Prova en PC local

Aquests proves han sigut realitzades en un SONY Vaio VGN-NW21EF amb la següent especificació:

- Ubuntu 10.4 32 bits
- 4gb RAM
- 2 processadors Pentium(R) Dual-Core CPU T4300 @ 2.10GHz

En aquest entorn s'ha instal·lat tots els components necessaris de la aplicació i amb la següent configuració:

- Ichnaea Software.
- Gestor de cues RabbitMQ.
- Sistema de cues per Ichnaea Software.
- Servidor Web Apache.
- Motor de dades MySQL.
- SMTP deshabilitat. Les comunicacions es capturen mitjançant el *profiler* de Symfony2.
- Memòria cau deshabilitada.

Creació i administració d'usuaris

S'han provat satisfactoriament totes les proves sense cap tipus de incidència. Les funcionalitats son extensions del paquet FOSUserBundle, que conté totes les funcionalitats provades per la comunitat.

Gestió de matrius

Creació de matrius Partim d'una matrius de dades reals proporcionada per Anicet XXX anomenada *Cypruss*. És una matriu composta per 103 mostres de 27 variables(columnes). La importació(creació del model de dades a partir d'un fitxer) triga 1226ms i la renderització 1066ms.

Esborrat de matrius La prova és un esborrat de matrius amb 4 entrenaments i dos prediccions ocupa 1153ms.

Esborrat d'un entrenament La prova d'un esborrat d'un entrenament amb dos prediccions triga 857ms.

Temps de resposta de les API JSON Restful Les llibreries de les interfícies enriquides son usades en les configuracions de matrius, tant de les normals com les de predicció. El temps mínim calculat ha sigut 407ms i el màxim 572ms.

Creació de matrius de predicció S'ha emprat una matriu de test per fer prediccions de 19 mostres. La creació del model de dades des d'un fitxer ha trigat 857ms i la renderització 855ms.

Rendiment d'Ichnaea Software en aquest entorn Aquest project no consisteix en l'estudi de rendiment d'Ichnaea Software. Però es interesant fer un estudi de temps en la realització dels processos.

Entrenament de *Cypruss* L'entrenament de la matriu *Cypruss*(mirar 6.2) ha tingut diferents mesures. S'han pres 3 mesures diferents: 1h 47min, 2h 15min i 1h 55min. Varia segons la carrega de la CPU, ja que requereix una alta carrega de CPU.

Predicció de *Cypruss* La predicció de la matriu de test(mirar 6.1) ha tingut diferents mesures. S'han pres 3 mesures diferents: 35min, 57min i 1h 5min, aproximadament. Varia segons la carrega de la CPU, ja que requereix una alta carrega de CPU.

6.2 Prova en entorn distribuït

Les proves en un entorn distribuït s'han realitzat en un màquina virtual proporcionada per RdLab amb la següent especificació.

- 2 processadors QEMU Virtual CPU version 1.0 a2659Mhz
- 2Gb RAM
- Linux version 3.2.0-30-generic

En aquest entorn s'ha instal·lat:

- Sense Ichnaea Software.
- Sense gestor de cues RabbitMQ.
- Sense sistema de cues per Ichnaea Software.
- Servidor Web Apache.
- Motor de dades MySQL en un altre servidor
- SMTP habilitat.
- Memòria cau habilitada.

No s'han pogut executar prediccions ni executar entrenaments.

Creació i administració d'usuaris

S'han provat satisfactoriament totes les proves sense cap tipus de incidència. Les funcionalitats son extensions del paquet FOSUserBundle, que conté totes les funcionalitats provades per la comunitat.

Gestió de matrius

Creació de matrius Partim de la mateixa matriu de dades reals proporcionada. La importació (creació) del model de dades a partir d'un fitxer triga 1826ms i la renderització 1466ms sense haver passat per memòria cau. Després de la segona execució la renderització triga 627ms.

Temps de resposta de les API JSON Restful Les llibreries de les interfícies enriquides son usades en les configuracions de matrius, tant de les normals com les de predicció. El temps mínim calculat ha sigut 216ms i el màxim ha sigut 297ms.

6.3 Errors coneguts

Creació d'entrenaments parcials

A l'hora de crear entrenaments existeix la possibilitat de crear entrenaments parcials, seleccionant les columnes que vols entrenar. Actualment Ichnaea Software utilitza un fitxer de configuració estàtic i està esperant un conjunt de variables que si no es presenten dona un error de lectura del contingut de fitxers.

Capítol 7

Conclusions

En aquest capítol es descriu la evolució temporal, la gestió del projecte, la estimació econòmica i les possibles millores.

7.1 Metodologia àgil

Per la realització del projecte s'han emprat un metodologia agile i aprofitant tècniques i artefactes de l'SCRUM.[?] S'han fet iteracions quinzenals amb el "product owner" canviant requisits, definint noves especificacions en cada una de les reunions, definint nous requisits i fent petites *demos* dels avanços. Els principis bàsics de la metodologia àgil són[16]:

- Els individus i les seves interaccions per sobre dels processos i les eines.
- El programari que funciona per sobre de la documentació exhaustiva.
- La col·laboració amb el client per sobre de la negociació de contractes.
- La resposta davant del canvi per sobre de seguir un pla tancat.

En el cas d'aquest projecte s'ha usat l'artefacte principal "Backlog" mitjançant histories de usuaris. Les historia de usuari son una una representació d'un requisit de software escrit en una o dos frases. En aquest cas:

Com **rol** vull fer **alguna cosa** per **obtenir benefici**

7.1.1 Backlog

Ens hem basat en les histories d'usuaris amb el següent *backlog* puntuant segons la serie de Fibonnaci.[17]

7.2 Estimació econòmica

7.3 Millores en futures versions

- Refactorització del codi
- Personalització dels perfils d'usuari: formats de dades, dates, idiomes en el cas de internalització
- Ampliació de les API Restful
- Depuració en els exploradors més habituals. La aplicació està depurada per Mozilla Firefox i Google Chrome. Però no ha sigut per Internet Explorer o Safari.
- Depuració en entorn distribuït del sistema de cues. Encara que la aplicació ha sigut desenvolupada per ser distribuïda i separada del sistema de cues, no ha sigut possible provar-la en un entorn més complexe i distribuït.
- Sistema de notifikacions. Un sistema més elaborat de notifikacions. Per exemple, notifikacions quan un "training" ha acabat o enviar
- Sistema de projectes. La aplicació no ha sigut contemplada per ser més col.laborat-iva. Hauria de ser més concurrent contra alguns recursos, per exemple diversos editors d'una matriu al mateix temps.
- Sistema de invitacions. Un sistema per invitar col.laboradors tant siguin de la plataforma com si no.
- Implementar tests automàtics.

Capítol 8

Bibliografia

Bibliografia

- [1] Aitor Pérez Pérez "ICHNAEA 2.0: a software for microbiology modelling" pp. 7-34, Feb. 2014
- [2] D. Sánchez "A Software System for the Microbial Source Tracking Problem" 2012
- [3] "Learn Symfony - Symfony" [Online]. Disponible a "<http://symfony.com/com/doc/current/index.html>"
- [4] "JSON API" [Online]. Disponible a "<http://jsonapi.org>"
- [5] "jQuery: The Write Less, Do More, JavaScript Library" [Online]. Disponible a "<http://jquery.com>"
- [6] "The Doctrine Project" [Online] Disponible a "<http://www.doctrine-project.org>"
- [7] "MySQL :: The world's most popular open source database" [Online] Disponible a "<http://www.mysql.com>"
- [8] "Twig - The flexible, fast, and secure PHP template engine" [Online] Disponible a "<http://twig.sensiolabs.com>"
- [9] "Model-view-controller" [Online] Disponible a "<http://en.wikipedia.org/wiki/Model-view-controller>"
- [10] Fabien Potencier "What is Symfony2?" [Online] "<http://fabien.potencier.org/article/49/what-is-symfony2>"
- [11] "Advanced Message Queuing Protocol" [Online] "http://es.wikipedia.org/wiki/Advanced_Message_Queueing_Protocol"
- [12] "El patrón Repositorio (Repository Pattern)" [Online] "<http://barradevblog.wordpress.com/2013/04/23/el-patron-repositorio-repository-pattern-implementacion-practica-con-entity-framework/>"

- [13] "Mapeo objeto-relacional" [Online] "http://es.wikipedia.org/wiki/Mapeo_objeto-relacional"
- [14] Martin Fowler [Online] "<http://martinfowler.com/articles/injection.html>"
- [15] RabbitMQ [Online] "<https://www.rabbitmq.com/>"
agile [Online] "<http://agilemethodology.org/>"
- [16] [Online] "<http://agilemanifesto.org/iso/es/>"
- [17] [Online] "<http://scrummethodology.com/scrum-effort-estimation-and-story-points/>"
- [18] Matthew David "HTML5: Designing Rich Internet Applications (Visualizing the Web)"
- [19] A Fully Baked PHP Framework [Online] "<http://ellislab.com/codeigniter>"
- [20] Active record Pattern [Online] "http://en.wikipedia.org/wiki/Active_record_pattern"
- [21] Composer, Dependency Manager for PHP [Online] "<https://getcomposer.org/>"
- [22] FriendsOfSymfony/FOSUserBundle [Online] "<https://github.com/FriendsOfSymfony/FOSUserBundle>"
- [23] List of HTTP status code [Online] "http://en.wikipedia.org/wiki/List_of_HTTP_status_codes"
- [24] HTTP cookie [Online] "http://en.wikipedia.org/wiki/HTTP_cookie"
- [25] Session ID [Online] "http://en.wikipedia.org/wiki/Session_ID"

Apèndix A

Manual d'usuari

A.1 Casa de l'usuari

Per accedir la casa del usuari es pot anar accedint mitjançant el logo del menú situat a l'esquerra del menú.

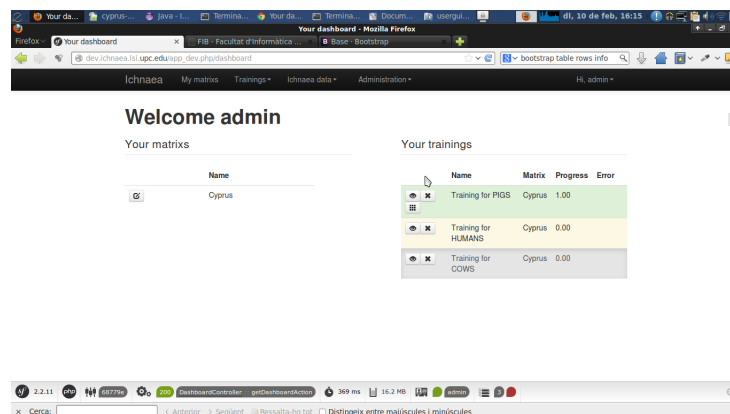


Figura A.1: Casa de l'usuari

A continuació descriurem cadascuna de les parts.

A.1.1 Llistat dels meus trainings pendents

Al llistat superior pots veure els trainings que has creat on::

- Nom de la matriu com a enllaç per veure la matriu
- Descripció del "training"
- Data de creació

- Nom de l'usuari que la creat
- Progrés: actualment Ichnaea no retorna estat del proces. Solament ens diu si ha acabat o no. Per tant els únics valors son 0.00 i 1.00.
- Status. Els possibles status son "pending"(no s'ha pogut enviar), "sent"(s'ha enviat a la cua) i "finished"(ha terminat)
-
- Operacions
 - ICONEYEOPEN es per anar a la pantalla de visualització del training.

Estat del trainings

En la figura es contempla els estats possibles:

- Color verd: training sense errors i predecible
- Color gris: training actualment corrent
- Color salmó: "training" no s'ha enviat per algun problema amb la cua o amb errors.

A.2 Variables

A.2.1 Veure les variables del sistema

Desde el menu "Ichnaea Data - View Variables", es poden veure totes les variables del sistema. Fent click a la icona de edició, es pot accedir a la interfície de configuració de la variable seleccionada.

A.2.2 Formulari de edició d'una variable

Desde la interfície de configuració es pot:

- Modificar la descripció
- Accedir a la vista per afegir una "Season set", amb la icona del signe de suma.
- Editar una "Season Set", amb ICONEDIT.
- Esborrar una "Season Set", amb la ICONTRASH.

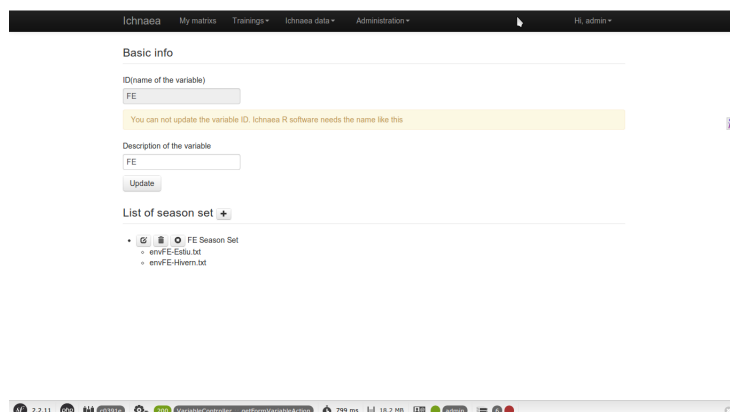


Figura A.2: Configuració de variable

A.2.3 Crear una "season set" d'una variable

Per tal de crear un conjunt de seasons per una variable:

- Llistar les variables per seleccionar una variable('Ichnaea Data - View variables')
- Editar la variable a la que es vol afegir la variable, fent click ICONEDIT
- Al apartat inferior "List of season set", clickar a la ICONPLUS.

Es poden donar un nom i afegir tants fitxers com es vulguin. S'ha de tenir cura perquè existeix la possibilitat de afegir varis fitxers amb una mateixa configuració (estació). Actualment Ichnaea solament processa estiu e hivern. Per més descripció d'aquesta interfície anar al punt A.2.4

A.2.4 Editar una "Season Set" d'una variable

Desde aquesta vista es pot:

- Editar el nom de la season set
- Eliminar simplement la relació amb una season amb la icona de la brossa.
- Eliminar un fitxer de la "Season Set" i la relació, amb la icona de la creu

S'ha de tenir cura perquè existeix la possibilitat de afegir varis fitxers a una mateixa estació. Actualment Ichnaea solament processa estiu e hivern.

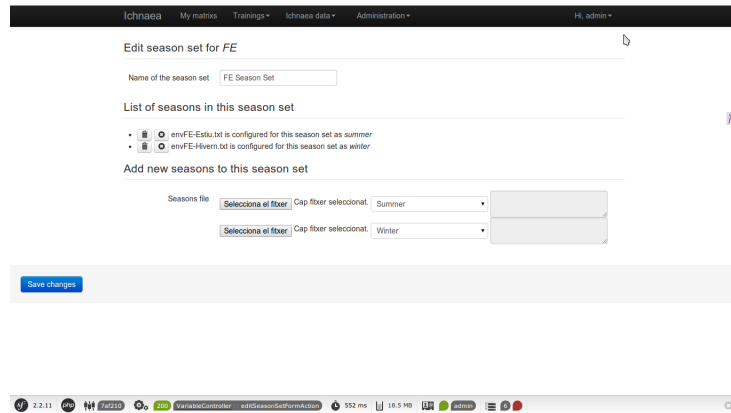


Figura A.3: Edició de una "Season Set"

A.2.5 Afegir una "season" a una "season set" existent d'una variable

Per tal de crear un conjunt de seasons per una variable:

- Llistar les variables per seleccionar una variable('Ichnaea Data - View variables')
- Editar la variable a la que es vol afegir la variable, amb ICONEDIT
- Editar, amb ICONEDIT, la "season set" del llistat "List of season set". Llegir el punt A.2.4

A.3 Matrius

A.3.1 Crear una matriu desde un fitxer csv o excel

Desde el menu superior "IchnaeaData - New matrix", es pot pujar una nova matriu en format csv o Microsoft Excel. El format csv es compatible amb les programaris de ofimàtica més habituals como Microsoft Excel o Libreoffice. El format de la matriu és important que sigui el següent.

Cel.la buida DATE	Alias de la columna	ORIGIN
Nom de la sample	Valor de la sample	Origen de la sample
S01-10-20	0,000145	Human

On:

- Alias de la columna: és un nom qualsevol per identificar la columna. Si el sistema conté una variable amb el mateix nom, automàticament li assignar aquesta variable amb una "season set" per defecte. Si l'alias de variable definit al full de càlcul és el mateix que el d'una variable del sistema, s'assigna a la variable i li assignar un conjunt de fitxers per defecte.
- Valor de la sample: és el valor de la mostra per la columna(variable)
- Nom de la sample: és un identificador de la mostra. La aplicació mapeja el contingut de subcadena amb:
 - * PL - POULTRY
 - * HM - HUMAN
 - * PG - PIG
 - * CW - COW
- Origen de la mostra: és una cadena de caràcters que especifica l'origen de la mostra. Solament es distingirà si a les capçaleres a la ultima columna s'especifica la paraula ORIGIN. Es OPCIONAL.
- Data de la mostra: és una data de calendari. Amb el format DIA(DD)/MES(MM)/ANY(YYYY). Per exemple 01/12/2014. Es OPCIONAL.

Important: l'ordre de les columnes és important. Si volem deixar la columna origen buida però posar dates, s'ha d'especificar igualment la columna i deixar-la buida. I l'origen de la mostra es resoldrà amb el mapejat abans explicat. En la pantalla, es pot seleccionar un fitxer csv i pujar'ho. Seguidament, es podrà establir la relació de la variable real de la columna i quin conjunt de fitxers per defecte usa. Mirar A.4.

A.4 Configurar una matriu

Per accedir a configurar una matriu, has d'anar a la teva pantalla de inici(mirar A.1). Desde la interfície de configuració es pot configurar:

- Donar un alias a la columna
- Associar una columna a una variable
- Seleccionar un conjunt de fitxers de la variable

- Donar nom a una mostra
- Donar una data a una mostra
- Donar un origen a un mostra
- Visualitzar missatges de validacions i notificacions
- Donar accés als usuaris per que puguin crear trainings.

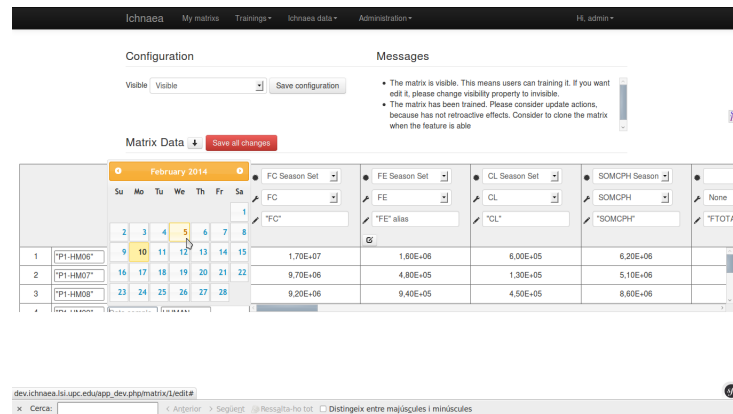


Figura A.4: Interfície de configuració de matrius

A.4.1 Alies de una columna

A la secció de les capçaleres, a la icona del llapis, es pot especificar un alies a la columna. Si es prem "Enter" o es canvia el focus, s'activa el botó de salvaguardat.

A.4.2 Especificar en una columna una variable i un conjunt de fitxers

A la secció de les capçaleres, a la icona de la clau anglesa, es pot seleccionar la variable del sistema. Automàticament, a la llista de dalt, es carrega la llista de "Seasons Set". Quan es selecciona un dels dos llistats, s'activa el botó de salvaguardat. No és obligatori donar-li una variable i una "Season Set".

A.4.3 Cambiar la visualització

A la secció de configuració, es pot canviar la visibilitat. Si la matriu és invisible, els usuaris no poden crear "trainings". Per guardar els canvis, s'ha de pitjar el butó "Save configuration".

A.4.4 Visualitzar missatges

Existeixen diverses restriccions i missatges:

- Notificació de visibilitat: una matriu visible es entrenable.
- Notificació de matriu amb trainings creats. Una modificació crea una incoherència amb aquests trainings ja que no serà la mateixa matriu.
- Notificació d'origens. Les mostres necessiten obligatoriament uns orígens.

A.4.5 Actualitzar una dada d'una fila i d'una columna

Es poden actualitzar les dades d'una mostra i d'una

A.5 Clonar una matriu

Desde el menú "Ichnaea Data - View Matrix", podem accedir al llistat de variables del sistema. Amb la icona etiquetada com "Clone the matrix", podem clonar una matriu sencera configurada. No es copien els trainings. Fent click a la icona de reload, anem al formulari que suggereix un nom per identificar-la. Acceptant, es clona la matriu i anem a la interfície de configuració. Mirar A.4.

A.6 Crear un training d'una matriu

Per crear un training s'ha de accedir al menú superior "Trainings - Create a training". Desde el llistat de matrius del sistema, amb la icona ICONROAD, es pot accedir al formulari de creació de trainings.

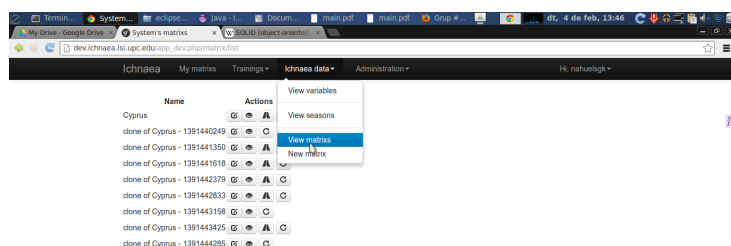


Figura A.5: Llistat de matrius

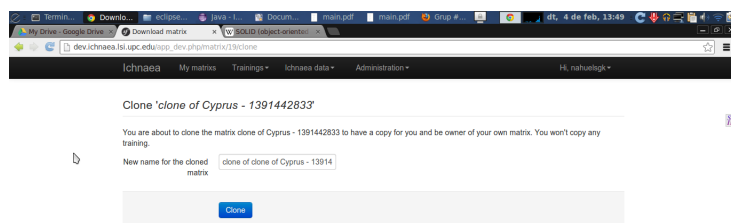


Figura A.6: Llistat de matrius

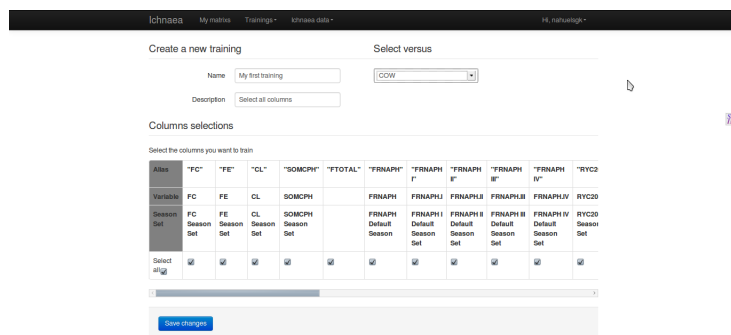


Figura A.7: Llistat de matrius

- Part esquerra superior: El training conté un nom i una descripció. Es poden seleccionar quines columnes vols entrenar de la matriu.
- Part dreta superior: Desplegable per seleccionar un dels orígens disponibles. El origen-versus, es un llistat de la variable origen de la matriu. Si es selecciona el valor "All versus all", el training serà tots contra tots. Si es selecciona un origen concret, el training es farà aquest origen contra els altres. Actualment Ichnaea no suporta aquesta part però en el futur està planificat que ho farà.
- Selecció de columnes. Selecció de columnes que vols entrenar.

Si la creació és correcta, les dades s'enviaran a la cua de processos i la aplicació es redirigirà la pantalla de visualització de trainings.

A.6.1 Simular un training de la matriu Cyprus

Actualment la aplicació Ichnaea i el sistema de cues no està implantat. Tenim la opció de tenir una matriu entrenada en una altra plataforma per poder fer proves amb les interfícies de prediccions. Pendent d'implementació.

A.7 Visualitzar un training

Desde la casa de l'usuari (mirar A.1), es pot veure els teus trainings i en quin estadi es troben. Amb la icona ull", pots accedir a visualitzar la informació del training.

A.7.1 Problemàtiques de la creació de trainings

Error en el enviament

Actualment Ichnaea Software i el sistema de cues no està implantat. Per defecte, la creació donarà error. Per tant, es pot utilitzar la simulació de trainings. A banda d'això, tenim predifinitos un conjunt de situacions que a continuació descrivim.

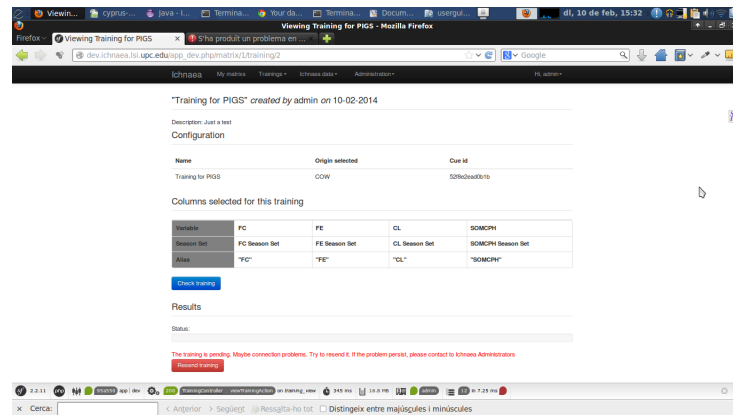


Figura A.8: Training que es pot enviar a la cua

A.8 Crear una matriu de predicció

Desde la casa de l'usuari, es pot veure els teus trainings i en quin estadi es troben. Amb la icona "Quadrats"

A.9 Crear una predicció

Desde la casa de l'usuari(mirar A.1) es pot crear una predicció d'un training. Seleccionant la icona "quadricules" de un trainig correcte(en color verd), es pot crear un matriu de predicció.

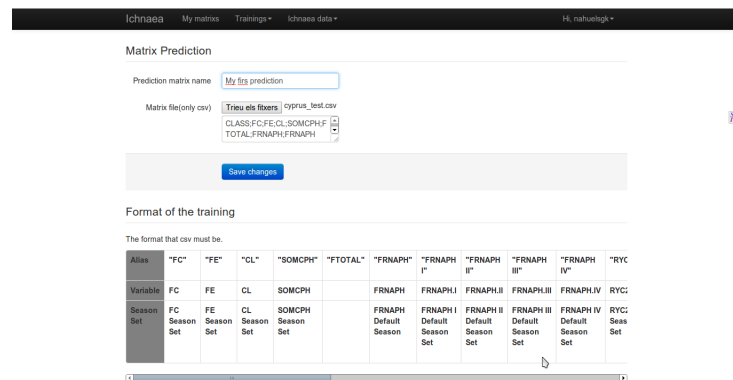


Figura A.9: Exemple de creació de predicció

A part superior és pot seleccionar un fitxer per pujar la matriu per predir. A la part inferior es pot veure les columnes que el training té

seleccionades. La matriu en format csv ha de tenir el format indicat per la part inferior. En breu es podrà descarregar una template per tenir el template i poder simplement omplir els valors:

Cel.la buida	Alias de la columna	...	ORIGIN
Nom de la sample	Valor de la sample	...	Origen de la sample
S01-10-20	0,000145	...	Human

Seguidament es pot visualitzar

A.10 Actualitzar una matriu de predicció

A.11 Llistar les meves prediccions

Desde el menu superior "Prediction - My predictions" pots llistar les teves prediccions.

A.12 Visualitzar una matriu de predicció

A.12.1 Actualitzar una matriu de predicció

A.12.2 Executar una predicció de una matriu de predicció

Apèndix B

Manual d'administrador

B.1 Llistar usuaris del sistema

Desde el menu superior "Administration - Users", els usuari poden llistar els usuaris del sistema.

B.2 Canviar el grup de l'usuari

Accedir al B.1 i seleccionar la icona de edició de un usuari. L'administrador accedirà a un menu on pot canviar a quin grup pertany l'usuari.

B.3 Comprovar cua

L'administrador pot accedir a "Administration - Check queue" a les eines per comprovar possibles problemàtiques amb les cues.