

**FACULTAD**  
**INFORMÁTICA MATEMÁTICA**

## **JCLAL-WEB**

### Manual de Usuario

Autor: Ramón Alejandro Valdés Ochoa

HOLGUÍN 2019



# Tabla de contenido

<b>1</b>	<b>Introducción.....</b>	<b>4</b>
1.1	JCLAL-WEB .....	6
<b>2</b>	<b>Instalación.....</b>	<b>8</b>
2.1	Instalación del servidor web de la aplicación. ....	8
2.1.1	NodeJS .....	8
2.1.2	MongoDB.....	8
2.1.3	JCLAL.....	8
2.1.4	JCLAL-WEB .....	8
<b>3</b>	<b>Uso de JCLAL-WEB .....</b>	<b>9</b>
3.1	Sign Up .....	10
3.2	Profile y Group Manage.....	11
3.3	Configuración de un experimento.....	15
3.4	Code Editor .....	19
3.5	Experiments.....	21
<b>4</b>	<b>Casos de estudio .....</b>	<b>24</b>
4.1	Caso de estudio .....	24
	<b>Referencias bibliográficas .....</b>	<b>28</b>

## Índice de Figuras

Figura 1 - Interfaz inicial o Dashboard .....	9
Figura 2 - Sign Up .....	10
Figura 3 - User Profile .....	11
Figura 4 - Group Manage (cuando el usuario aún no pertenece a algún grupo) .....	12
Figura 5 - User Profile mostrando tarjeta con miembros del grupo .....	13
Figura 6 - Group Manage .....	13
Figura 7 - Group Manage, promover miembro o eliminar .....	14
Figura 8 - Experiment Configuration. Process .....	15
Figura 9 - Experiment Configuration. Data Sets .....	16
Figura 10 - Experiment Configuration. Algorithms .....	17
Figura 11 - Experiment Configuration. Query Strategy .....	17
Figura 12 - Experiment Configuration. Listeners .....	18
Figura 13 - Experiment Configuration. Spark .....	19
Figura 14 - Code Editor. Tema Claro .....	20
Figura 15 - Code Editor. Tema oscuro y Botones .....	21
Figura 16 - Experiments. Resultados .....	22
Figura 17 - Experiments. Historial y Reporte .....	23
Figura 18 - Configurando experimento. Proceso .....	25
Figura 19 - Accediendo a "Experiment Configuration" .....	25
Figura 20 - Configurando experimento. Conjunto de datos .....	26
Figura 21 - Configurando experimento. Algoritmo .....	26
Figura 22 - Configurar experimento. Estrategia de consulta .....	27

## Índice de Tablas

Tabla 1 - Caso de Estudio .....	24
---------------------------------	----

# 1 Introducción

El estudio de la Inteligencia Artificial (IA) se divide en varias ramas dentro de las que se encuentra el **Aprendizaje Automático** (*Machine Learning*, ML, por sus siglas en inglés) en la que se incluye el desarrollo de algoritmos y técnicas que permiten al sistema aprender de forma automática basado en análisis de casos que generen experiencia previa. Para su funcionamiento es necesario el análisis de grandes volúmenes de datos para lo cual se utilizan distintos tipos de algoritmos que, en función a su funcionamiento se clasifican en su mayoría como **aprendizaje supervisado** (SL, por sus siglas en inglés, *Semi-Supervised Learning*), **aprendizaje no supervisado** y **semi-supervisado** (SSL, por sus siglas en inglés, *Supervised Learning*)[1].

Para la clasificación de nuevos casos se precisa de un modelo predictivo lo suficientemente preciso para sustituir a un humano por lo que es necesario contar con una amplia base de casos previamente etiquetados para la etapa de entrenamiento. Esta tarea se realiza por expertos humanos por lo que se convierte en un proceso costoso e ineficiente. Debido a lo poco práctico de este proceso surgen otras técnicas como el aprendizaje semi-supervisado y el aprendizaje activo<sup>1</sup> (AL, por sus siglas en inglés, *Active Learning*).

Con el AI se intenta mejorar la precisión de un clasificador disminuyendo la cantidad de ejemplos previamente etiquetados que se necesitan, permitiéndole al algoritmo seleccionar los datos desde donde aprende y reduciendo el volumen necesario en el proceso de entrenamiento para lograr un mejor rendimiento con un menor costo de entrenamiento. Está motivado por muchos problemas modernos de ML donde los datos pueden ser abundantes, pero las etiquetas son escasas o costosas de obtener.[2]

Los investigadores en el área de AL deben realizar experimentos para comparar entre el desempeño de los algoritmos y técnicas antes de su aplicación, lo que suele consumir bastante tiempo. En busca de facilitar el trabajo a los investigadores y por la falta de un *framework* que posea técnicas de esta área de estudio, se crea el *framework* JCLAL<sup>2</sup> (*Java Class For Active Learning* por sus siglas en inglés) para la experimentación y desarrollo de algoritmos de AL.

JCLAL es un software de código abierto para que los investigadores y los usuarios finales desarrollen métodos de AL. JCLAL incluye las estrategias de consulta más relevantes que se han propuesto en paradigmas de aprendizaje de etiqueta única

---

<sup>1</sup> Se le conoce también como aprendizaje por consulta o diseño experimental óptimo en la literatura de estadística

<sup>2</sup> <http://sourceforge.net/projects/jclal/files/>

y multi-etiqueta. Proporciona las interfaces, las clases y los métodos necesarios para desarrollar cualquier método de AL.[3]

La forma de realizar los experimentos en JCLAL es mediante la configuración de un archivo en formato XML donde se incluyen las etiquetas necesarias para la ejecución de los experimentos. Estas etiquetas hacen referencia a los principales elementos del flujo de AL, que son configurables mediante los paquetes de clases de JCLAL. Para ejecutar un experimento se deben usar distintos comandos desde la consola, que permitan al *framework* conocer los parámetros necesarios para desarrollar este proceso como son la dirección donde se encuentra el archivo de configuración, los métodos que se utilizaran en la experimentación o si se desea ejecutar el experimento en paralelo.[4]

En principio la experimentación con el *framework* era totalmente funcional; la interacción poco asistida en la configuración, la ejecución de los experimentos y comparación de los resultados obtenidos, provocaba aumento en el tiempo de trabajo, los investigadores debían dominar los paquetes de clases del *framework*, así como todas las etiquetas de XML<sup>3</sup> necesarias para realizar los experimentos, además la posibilidad de introducir errores de escritura en la configuración del XML y tener problemas en encontrar su origen podían dificultar mucho su uso. Era poco probable que usuarios con poca habilidad en el uso del *framework* o con conocimientos básicos de AL, fueran capaces de realizar los experimentos de forma rápida y sencilla.

A partir de estas deficiencias surge VisualJCLAL<sup>4</sup>; un entorno de usuario para el *framework* JCLAL que facilitaba mucho la configuración de nuevos experimentos. VisualJCLAL cuenta con un diseño de pestañas que permite configurar paso a paso un experimento de AL y mostrar los resultados posteriormente.

Aunque VisualJCLAL agilizó mucho el proceso de experimentación con JCLAL; algunos aspectos mejorables como la necesidad de tener instalada la maquina virtual de Java en el terminal donde se valla a ejecutar, la configuración algo confusa para los usuarios poco experimentados, así como la imposibilidad de realizar una gestión de usuario que brindaría posibilidades como el almacenamiento de historiales de experimentación y el trabajo colaborativo en general entre varios investigadores en un entorno de red o localmente; conllevarían a la siguiente evolución de esta potente herramienta, y, de esta forma permita su acercamiento a más usuarios y facilite aún más el proceso de aprendizaje y familiarización con JCLAL y el área de AL en general.

---

<sup>3</sup> XML (eXtensible Markup Language), "Lenguaje de Marcado Extensible", es un meta-lenguaje que permite definir lenguajes de marcas utilizado para almacenar datos en forma legible.

<sup>4</sup> [https://github.com/eperezp1990/jclal-develop/tree/visual\\_jclal](https://github.com/eperezp1990/jclal-develop/tree/visual_jclal)

Para darle solución a la problemática identificada se ha desarrollado una aplicación web que favorece el proceso de experimentación en el área de aprendizaje activo utilizando el *framework* JCLAL que permite la edición de el XML mediante un editor de código y el trabajo colaborativo entre varios investigadores. A continuación se presenta un manual de usuario para utilizar JCLAL-WEB, el cual cuenta con un conjunto de ejemplos que permitirán introducir a los investigadores en el uso de la solución propuesta. Este Manual de Usuario asume que el lector es conocedor del área de AL y del *framework* JCLAL antes mencionado, por lo que los conceptos básicos de estos no serán explicados.

## 1.1 JCLAL-WEB

La Aplicación web JCLAL-WEB permite realizar experimentos utilizando JCLAL para los tipos de datos single-label y multi-label de forma rápida y sencilla. Es libre y de código abierto. Está distribuido bajo la licencia GNU. Permite utilizar la mayoría de las opciones que brinda el *framework* JCLAL en su versión 2.0 incluyendo la experimentación en paralelo y distribuida, esta última utilizando el *framework* Apache Spark.

Brinda la posibilidad de utilizar escenarios, algoritmos y estrategias de consulta creados por el usuario mediante el uso de archivos de configuración en formato .props que son incluidos dentro de las opciones de elección de la interfaz. Permite visualizar los resultados de los experimentos, así como almacenarlos en archivos, reutilizando y agilizando algunas de las funcionalidades de JCLAL.

JCLAL-WEB se caracteriza por:

- **Usabilidad:** Cuenta con un diseño óptimo para agilizar el tiempo de respuesta del mismo. Los usuarios no necesitan conocer a fondo el funcionamiento interno del *framework* JCLAL.
  - **Aumento de funcionalidades:** la aplicación cuenta con un editor de código completamente funcional que permitirá la edición de archivos XML necesarios para la configuración.
  - **Colaboración:** la aplicación ha sido diseñada especialmente para facilitar su uso en ambientes colaborativos.
- **Rendimiento:** Ejecuta los experimentos de una manera eficiente y precisa. El tiempo de ejecución que necesita un experimento depende del conjunto de datos y del tipo de estrategia de consulta seleccionada.
- **Hardware:** Los requerimientos mínimos de hardware para JCLAL-WEB dependen del experimento a ejecutar en el servidor donde se encuentre montado el servidor web.
  - **Espacio:** no ocupa espacio en el disco duro local.

- **Consumo de recursos bajo:** las tareas que realiza el software no consumen recursos del terminal porque se realizan desde el servidor que esté ejecutando la aplicación.
- **Software:** Para el cliente solo es necesario contar con un navegador web.
  - **Actualizaciones inmediatas:** el usuario estaría usando siempre la última versión que se haya lanzado.
  - **Seguridad:** los datos de los usuarios se encuentran cifrados y seguros.
  - **Multiplataforma:** se pueden usar desde cualquier sistema operativo porque solamente es necesario tener un navegador.

## 2 Instalación

Para el uso de JCLAL-WEB desde un terminal no es necesaria la instalación del servidor web así que puede ir directamente al próximo [capítulo](#).

### 2.1 Instalación del servidor web de la aplicación.

Para la instalación del servidor web de JCLAL-WEB que permitirá el uso de la misma desde uno o varios terminales lo primero es descargar e instalar los recursos necesarios a continuación relacionados.

#### 2.1.1 NodeJS

Obtener la ultima versión de [NodeJS](#) descargando el instalador desde el sitio oficial o mediante un [gestor de paquetes](#).

#### 2.1.2 MongoDB

Obtener la ultima versión de [MongoDB](#) descargando el instalador desde el sitio oficial. Si tiene algún problema para instalar consulte la [documentación](#).

#### 2.1.3 JCLAL

Obtener el framework JCLAL descargando el archivo .jar desde [GitHub](#) o [SourceForge](#).

#### 2.1.4 JCLAL-WEB

Para obtener JCLAL-WEB deberá descargar el proyecto desde [GitHub](#). El archivo comprimido contendrá los archivos necesarios para ejecutar la aplicación una vez instalados los recursos necesarios para su despliegue.

### 3 Uso de JCLAL-WEB

Si ha montado el servidor web local podrá acceder desde su navegador a la dirección donde se encuentra alojado, por defecto sería <http://localhost:3000/>.

Lo primero que verá será la vista principal como se muestra en la Figura 1



Figura 1 - Interfaz inicial o Dashboard

Las opciones son:

- Inicio (Home): Permitirá volver a esta misma pagina desde cualquier parte de la aplicación.
- Configuración de experimento (Experiment Configuration): Permite acceder a la vista donde podrá configurar nuevos experimentos.
- Editor de código (Code Editor): Accede a la vista donde se encuentra el editor de código.
- Buscar (Search): Permite la búsqueda de contenido dentro de la aplicación.
- Login: Permite iniciar sesión de forma rápida
- Registrarse (Sign Up): Acceder al formulario que permite registrarse dentro de la aplicación.
- Características del framework JCLAL y JCLAL-WEB: Muestra las principales características de ambas tecnologías.

A continuación se describe el funcionamiento de la aplicación a partir de pulsar alguna de las opciones mencionadas.

### 3.1 Sign Up

Si se pulsa el botón "Sign Up" aparecerá una vista con un formulario donde podrá llenar los campos correspondientes a: Nombre (Name), Correo electrónico (Email Address) y contraseña (Password) y posteriormente pulsar el botón "Create" para registrarse en la aplicación. (Figura 2)

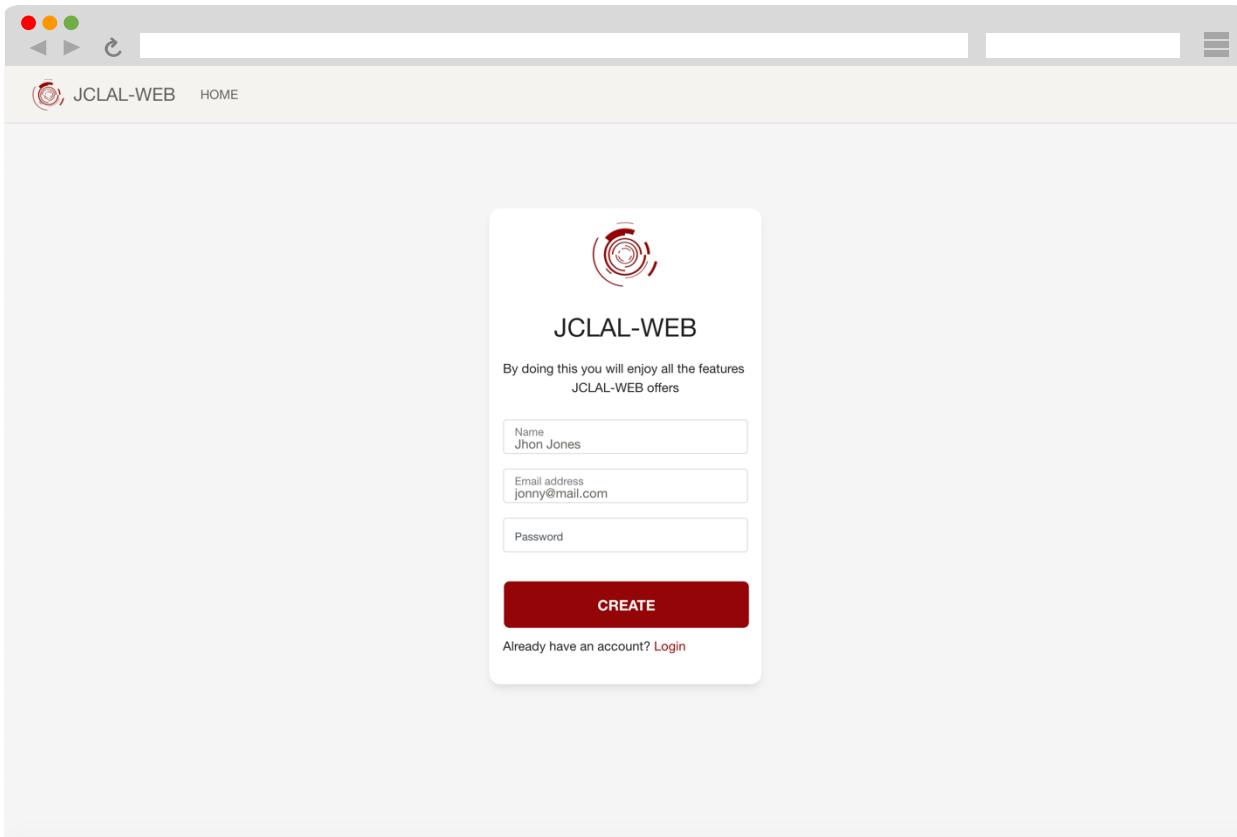


Figura 2 - Sign Up

Si ya esta registrado podrá acceder a una vista de login pulsando el vinculo "Login" desde la que podrá ingresar con su usuario y contraseña.

Nota: Para acceder a la vista de configuración de un experimento o al editor de código no es necesario iniciar sesión en la aplicación. Siendo requerido para visualizar los resultados de un experimento que se desee ejecutar y otras funcionalidades que permitirán el trabajo colaborativo explicadas más adelante.

## 3.2 Profile y Group Manage

Luego de haber iniciado sesión dentro de la aplicación podrá acceder a su perfil de usuario (Figura 3), acción que se realizará automáticamente después de culminar alguna de las dos acciones explicadas anteriormente, ya sea acceder o registrarse.

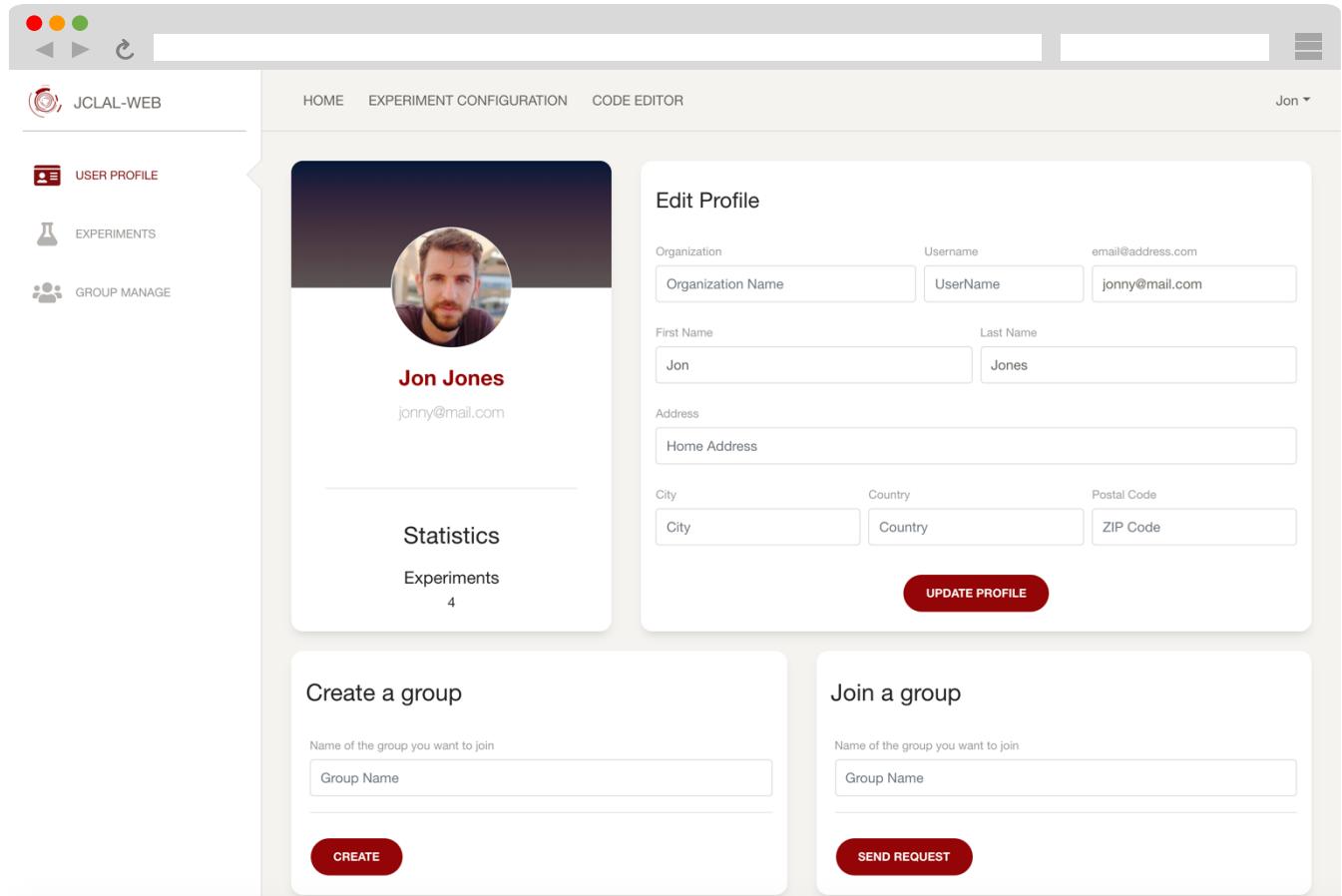


Figura 3 - User Profile

Desde esta vista podrá ver en la primera tarjeta a la izquierda un resumen de su perfil donde se encuentra la información pública que podrán ver otros usuarios de la aplicación. En la siguiente tarjeta a la derecha podrá agregar información a su perfil o editar la ya existente<sup>5</sup>.

JCLAL-WEB permite la a los usuarios formar grupos para trabajar en un ambiente colaborativo. Si aún no forma parte de ninguno de estos grupos; tanto en ésta vista como en la vista reservada para la gestión de grupos (Figura 4) se mostrarán dos tarjetas a través de las cuales podrá crear un grupo o unirse a uno ya existente.

<sup>5</sup> La información que no se muestra en la tarjeta de resumen no será pública en ningún momento.

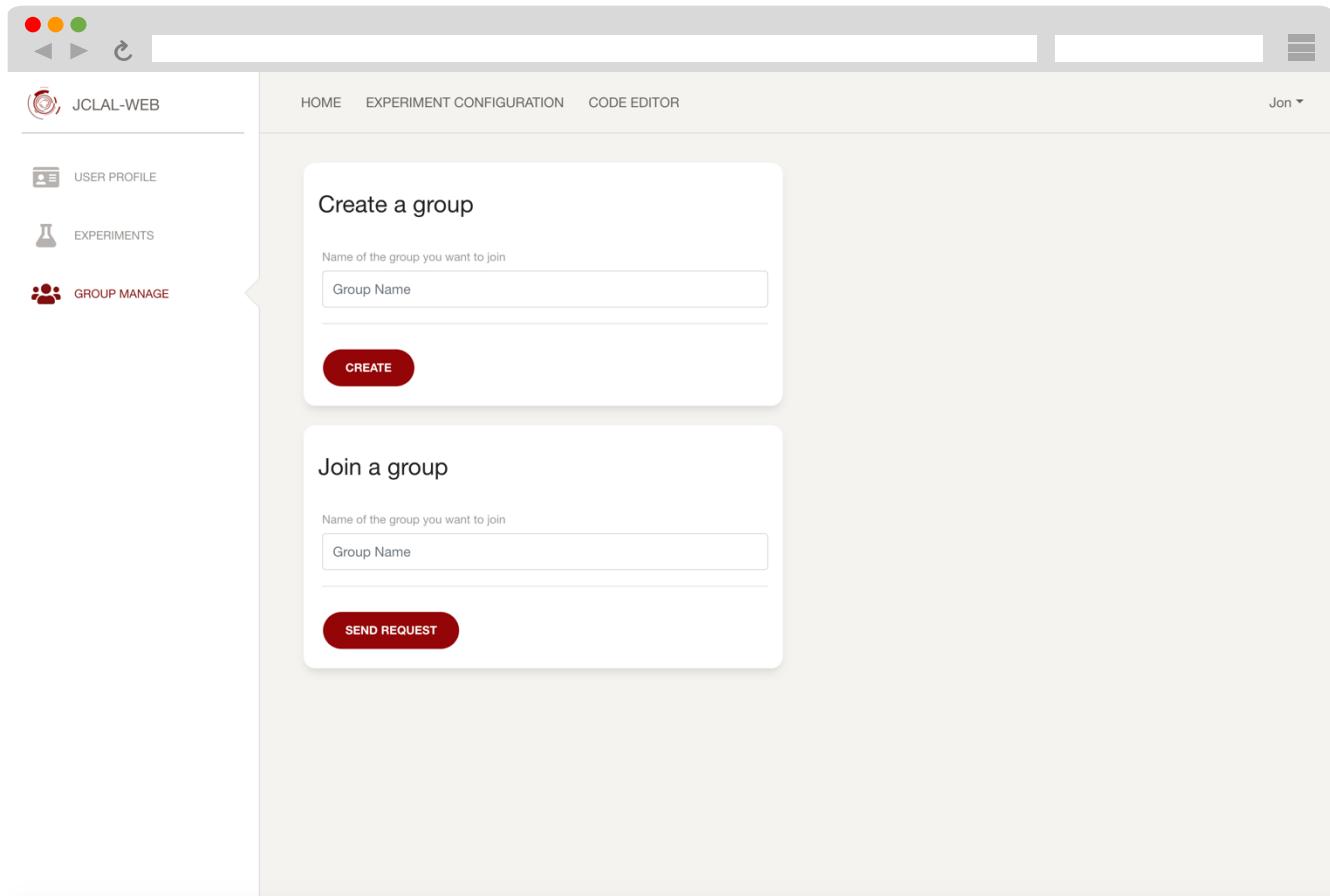


Figura 4 - Group Manage (cuando el usuario aún no pertenece a algún grupo)

Una vez que se halle en un grupo en la vista "User Profile" aparecerá una tarjeta con los miembros del grupo al que pertenece donde conocerá su estado (Conectado, Ocupado o Desconectado) y tendrá la posibilidad de enviarles un email de forma rápida pulsando el botón con el icono en forma de carta que se encuentra a la derecha de cada uno como lo muestra la Figura 5.

En la vista "Group Manage" ahora se mostrara en la primera sección el nombre del grupo y debajo una serie de tarjetas con información de los miembros que lo componen desde las cuales podrá acceder al perfil público de cada uno de ellos (Figura 6).

**Jon Jones**  
jonny@mail.com

**Statistics**  
Experiments 4

**Edit Profile**

Organization: Organization Name

Username: UserName

email@address.com

jonny@mail.com

First Name: Jon

Last Name: Jones

Address: Home Address

City: City

Country: Country

Postal Code: ZIP Code

**UPDATE PROFILE**

**Group Members**

- George St-Pierre (Offline)
- Anderson Silva (Available)
- Daniel Cormier (Busy)

Figura 5 - User Profile mostrando tarjeta con miembros del grupo

**InfoRec Research Team**

Profile	Name	Role	Action
	Jon	Team Leader	<a href="#">VIEW PROFILE</a>
	George St-Pierre	User	<a href="#">VIEW PROFILE</a>
	Anderson Silva	User	<a href="#">VIEW PROFILE</a>
	Daniel Cormier	User	<a href="#">VIEW PROFILE</a>

**Search Users To Join**

Search...

The results will be shown here

**SEARCH**

**Comunicate With Your Group**

Write your message

**SEND**

Use Slack Or The Messaging App That You Prefer

Figura 6 - Group Manage

En un grupo cada miembro tiene un rol asignado (por defecto "User") que puede ser modificado por el creador del grupo. Estos roles no están sujetos a ningún estándar o modelo, de esta forma cada grupo tendrá su propia jerarquía. Esta función solo está disponible para los creadores de un grupo. Se podrá hacer uso de esta funcionalidad al pasar el cursor por encima de alguna de las tarjetas de usuario como se muestra en la Figura 7. Al hacerlo aparecerán dos botones; el verde abrirá un input donde podrá escribir el rol que desea asignar a este miembro y el rojo servirá para expulsarlo del grupo. Es importante que conozca que si decide pulsar el botón rojo en su propia tarjeta el grupo quedará eliminado automáticamente y la información relacionada con este ya no será accesible.

The screenshot shows the JCLAL-WEB Group Management interface. The main content area displays the 'InfoRec Research Team' with four members: Jon (Team Leader), George St-Pierre, Anderson Silva, and Daniel Cormier. Each member has a circular profile picture and a 'VIEW PROFILE' button below it. To the left of the main content, there is a sidebar with navigation links: 'USER PROFILE', 'EXPERIMENTS', and 'GROUP MANAGE' (highlighted in red). Below the main content, there are two side panels. The left panel is for searching users to join, featuring a search bar and a 'SEARCH' button. The right panel is for communicating with the group, showing a message input field and a 'SEND' button, along with icons for Slack and other messaging platforms.

Figura 7 - Group Manage, promover miembro o eliminar

Debajo de esta sección se encuentra a la izquierda una tarjeta a través de la cual es posible realizar búsquedas de otros usuarios registrados en la aplicación para de esta forma poder agregarlos al grupo. Al hacer esto, al usuario se le enviará una solicitud que deberá confirmar para de esta forma quedar unido al grupo o rechazarla si no lo desea. Tenga en cuenta que un usuario solo puede pertenecer a un grupo, ya sea creador o no del mismo.

A la derecha arriba aparece una sección desde la que un usuario podrá enviar un mensaje de difusión que podrán ver los demás miembros del grupo la próxima vez que inicien sesión en la aplicación. Ya que esta es la única forma de comunicación interna que posee JCLAL-WEB por el momento; la siguiente sección sugiere el uso de Slack u otra aplicación que permita la comunicación entre los miembros del grupo.

### 3.3 Configuración de un experimento

En la vista "Experiment Configuration" aparecerá una interfaz de configuración donde se podrá configurar todos los parámetros necesarios para la ejecución de los experimentos. En la Figura 8 aparece la interfaz de configuración con la sección Process (Proceso) seleccionada donde se configuran los parámetros del Método de Evaluación y el Método de Muestreo.

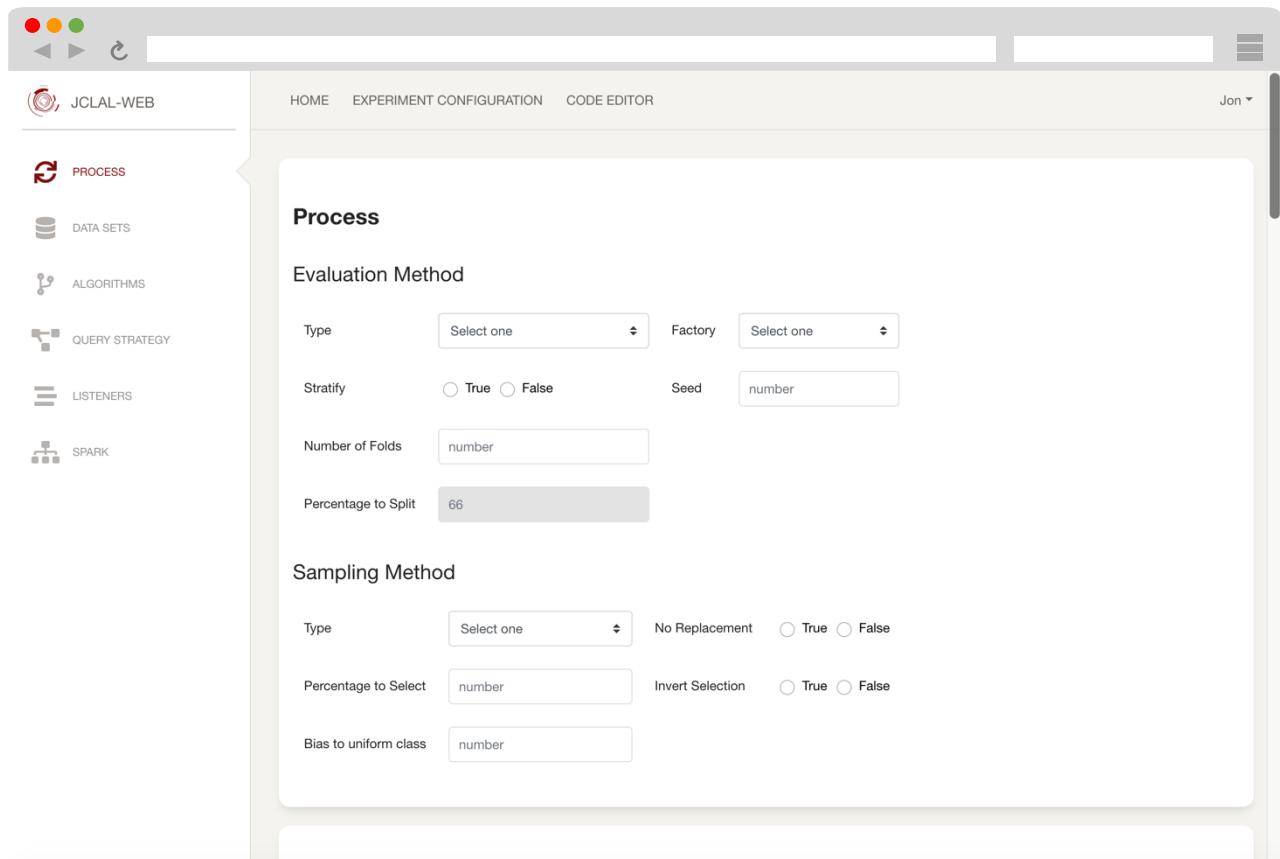


Figura 8 - Experiment Configuration. Process

Haciendo scroll podemos acceder a la sección correspondiente al Dataset (conjunto de datos) que se puede ver en la Figura 9 donde se puede elegir el conjunto de datos, el tipo de dataset y el índice de la clase. Además, si se selecciona un tipo de dataset correspondiente a la distribución utilizando Spark se

debe seleccionar el batch () de instancias a enviar, o en otro caso si se selecciona uno correspondiente a Mulan se debe añadir la ruta al archivo XML con la clase del dataset.

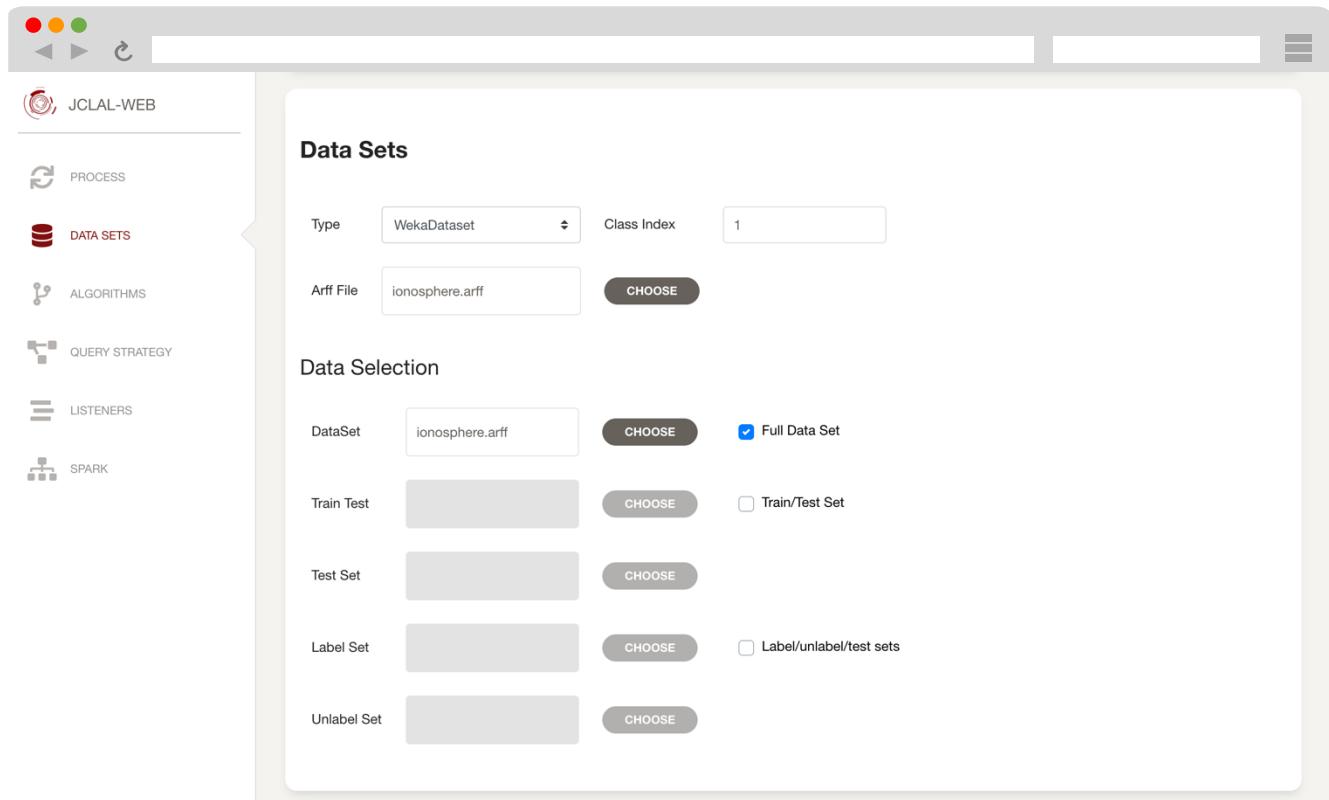


Figura 9 - Experiment Configuration. Data Sets

La próxima sección en la interfaz de experimentación corresponde a la tarjeta Algorithm (Algoritmo) que se puede observar en la Figura 10, donde se puede configurar datos relacionados al algoritmo de Aprendizaje Activo en general como son el propio algoritmo, el escenario, el oráculo y los criterios de parada.

A continuación la siguiente sección es la correspondiente a la Estrategia de Consulta (Query Strategy) que se puede apreciar en la Figura 11. En esta se puede configurar los parámetros correspondientes como el tipo de estrategia de consulta que se desea utilizar (si es single-label o multi-label), la estrategia de consulta en correspondencia con el tipo elegido con anterioridad, también si se desea paralelizar la ejecución del experimento y escoger la clase

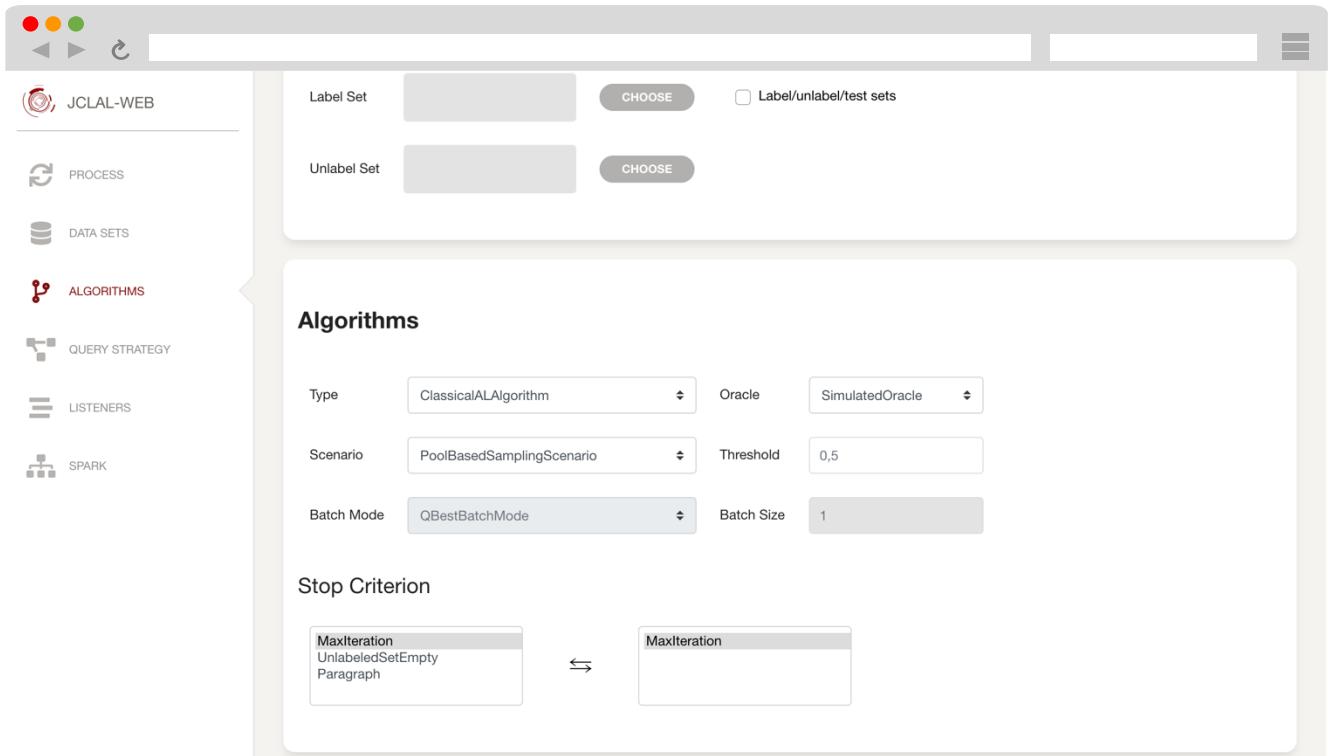


Figura 10 - Experiment Configuration. Algorithms

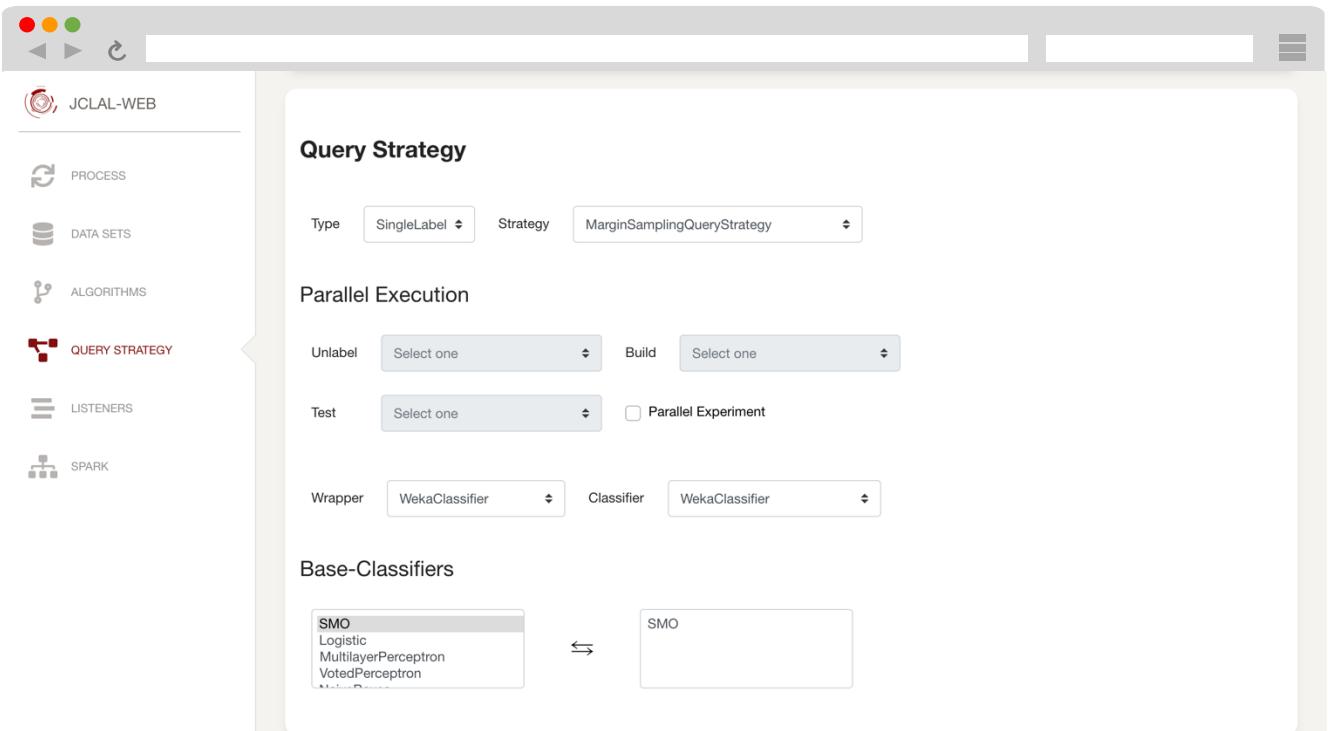


Figura 11 - Experiment Configuration. Query Strategy

Otra sección que se puede configurar opcionalmente (no es necesario configurar esta para una ejecución normal de un experimento) es la correspondiente a los Listeners donde se puede escoger el tipo de listener que se desea utilizar y configurar sus parámetros como se aprecia en la Figura 12.

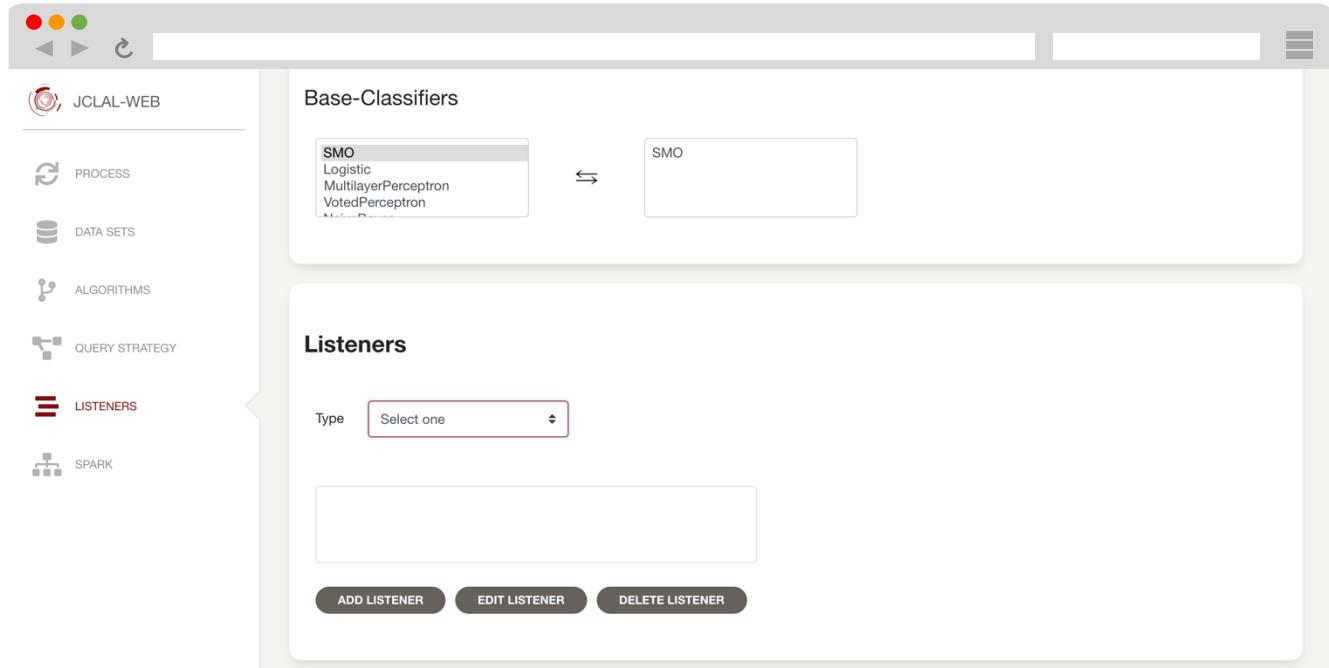


Figura 12 - Experiment Configuration. Listeners

Por último, otra sección opcional es la correspondiente a la experimentación distribuida utilizando Spark, ver Figura 13. En esta se debe seleccionar si se desea distribuir el experimento y agregar algunos de los parámetros de configuración propios de Spark como son el nombre que se le asignara a este proceso, el master de Spark, las variables de entorno y las librerías.

Habiendo concluido con la configuración de un experimento tenemos al final tres botones:

- “Execute Experiment” (Ejecutar Experimento): se ejecutará el experimento que hemos configurado.
- “Save” (Salvar): Salvar la configuración del experimento
- “Load” (Cargar): Cargar la configuración de un experimento previamente guardada.
- “Export” (Exportar): Exportar la configuración en formato XML

(Nota): Para realizar cualquiera de estas acciones excepto exportar, es necesario iniciar sesión en la aplicación.

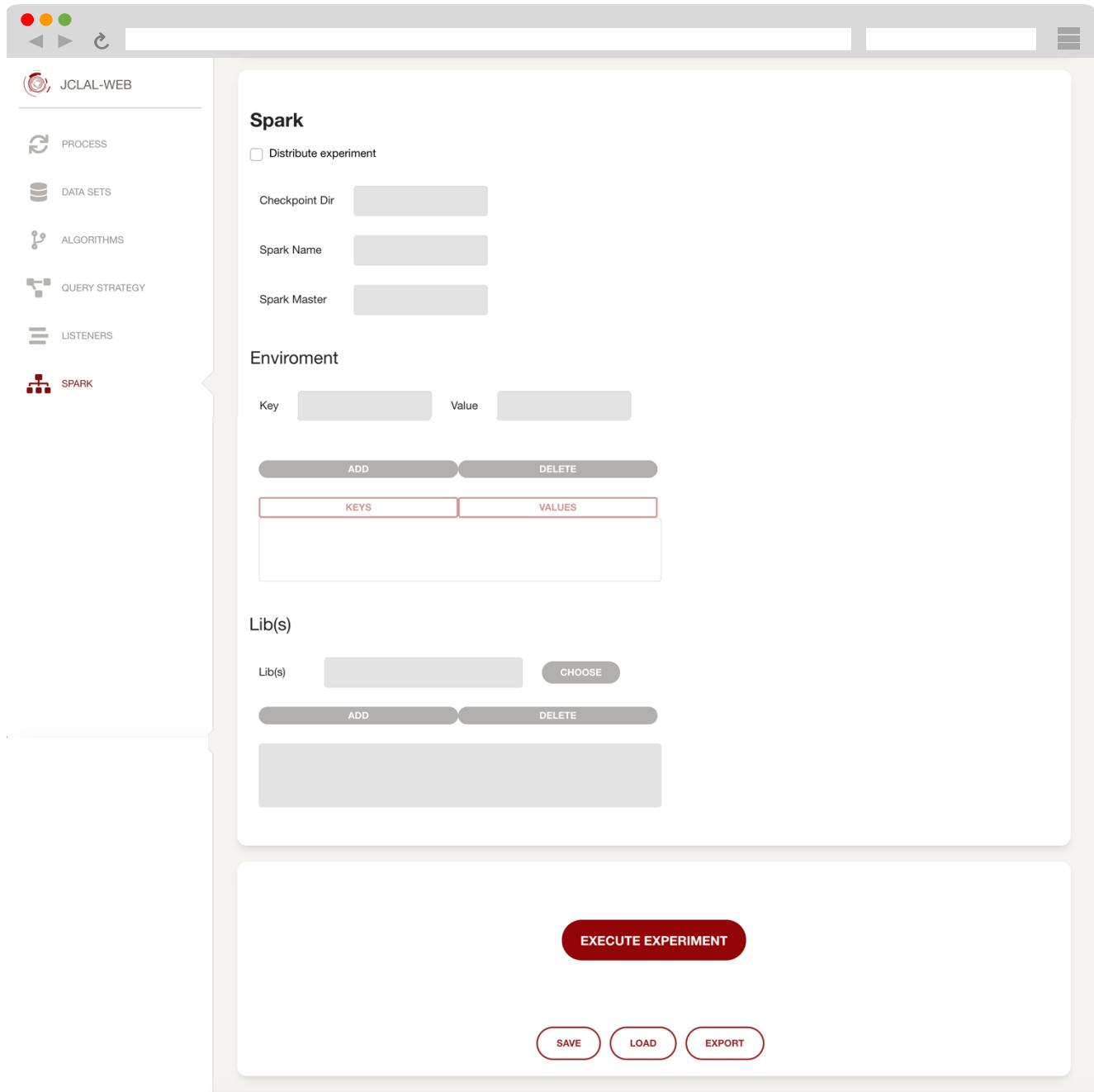
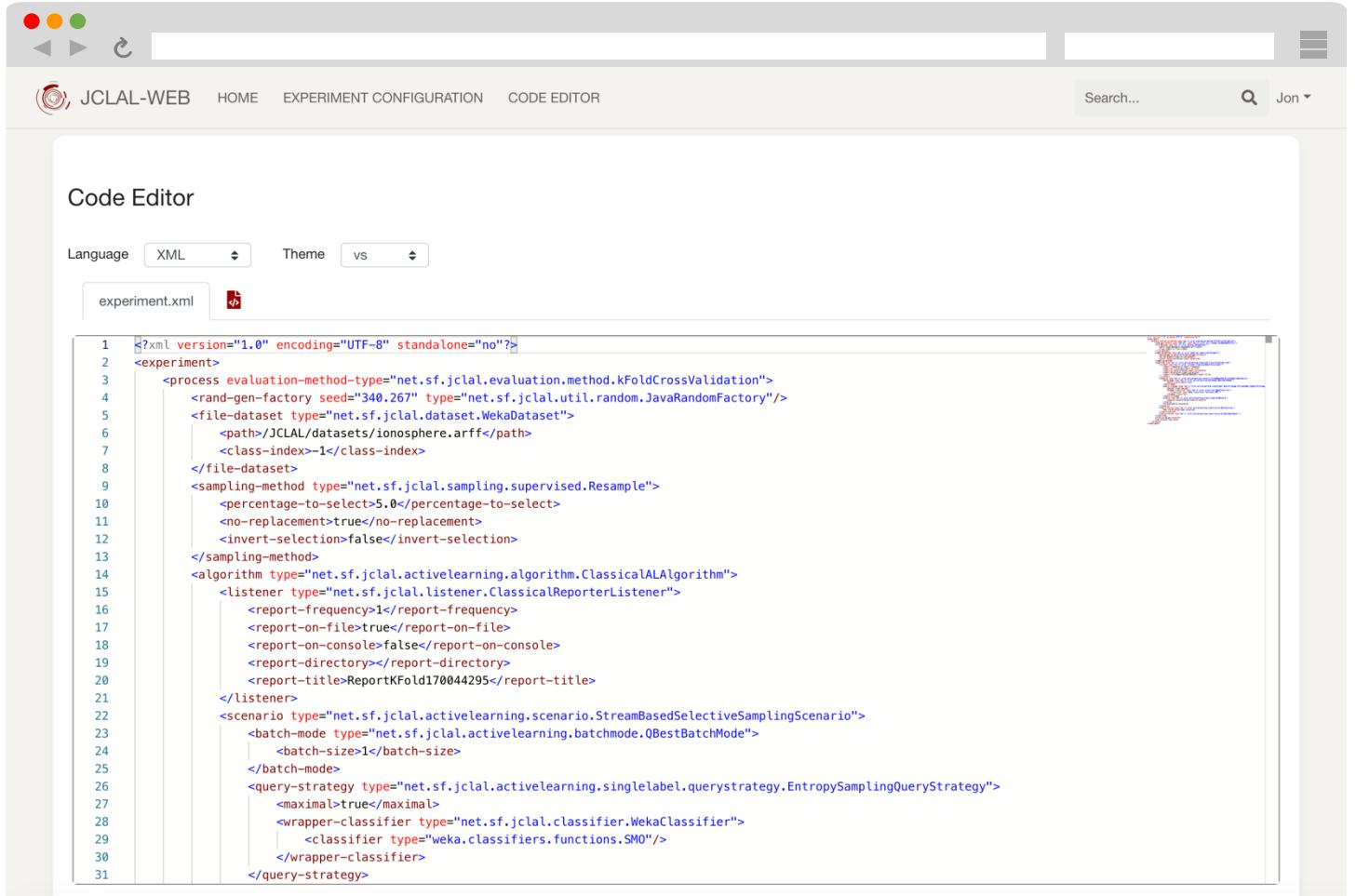


Figura 13 - Experiment Configuration. Spark.

### 3.4 Code Editor

JCLAL-WEB Cuenta con un editor de código completamente funcional donde podrá modificar el archivo .XML que se genere en la vista de configuración y ya que esta relación es directamente proporcional los cambios efectuados en el documento también serán reflejados en la vista de configuración.

Dentro de las funcionalidades que encontramos en la parte superior de la vista del editor está Lenguaje (por defecto seleccionado en XML) que permite cambiar entre varios lenguajes de programación<sup>6</sup> para así crear o modificar archivos en otros formatos y para comodidad del usuario este también podrá cambiar entre dos temas; claro y oscuro, según lo prefiera. Ver Figuras 14 y 15.



The screenshot shows the JCLAL-WEB interface with the 'Code Editor' tab selected. The title bar includes the logo, 'JCLAL-WEB', 'HOME', 'EXPERIMENT CONFIGURATION', 'CODE EDITOR', 'Search...', and a user 'Jon'. The main area is titled 'Code Editor' and contains a code editor with the following XML content:

```

1 <?xml version="1.0" encoding="UTF-8" standalone="no"?>
2 <experiment>
3   <process evaluation-method-type="net.sf.jclal.evaluation.method.kFoldCrossValidation">
4     <rand-gen-factory seed="340.267" type="net.sf.jclal.util.random.JavaRandomFactory"/>
5     <file-dataset type="net.sf.jclal.dataset.WekaDataset">
6       <path>/JCLAL/datasets/ionosphere.arff</path>
7       <class-index>-1</class-index>
8     </file-dataset>
9     <sampling-method type="net.sf.jclal.sampling.supervised.Resample">
10       <percentage-to-select>5.0</percentage-to-select>
11       <no-replacement>true</no-replacement>
12       <invert-selection>false</invert-selection>
13     </sampling-method>
14     <algorithm type="net.sf.jclal.activelearning.algorithm.ClassicalALAlgorithm">
15       <listener type="net.sf.jclal.listener.ClassicalReporterListener">
16         <report-frequency>1</report-frequency>
17         <report-on-file>true</report-on-file>
18         <report-on-console>false</report-on-console>
19         <report-directory></report-directory>
20         <report-title>ReportKFold170044295</report-title>
21       </listener>
22     <scenario type="net.sf.jclal.activelearning.scenario.StreamBasedSelectiveSamplingScenario">
23       <batch-mode type="net.sf.jclal.activelearning.batchmode.QBestBatchMode">
24         <batch-size>1</batch-size>
25       </batch-mode>
26       <query-strategy type="net.sf.jclal.activelearning.singlelabel.querystrategy.EntropySamplingQueryStrategy">
27         <maximal>true</maximal>
28         <wrapper-classifier type="net.sf.jclal.classifier.WekaClassifier">
29           <classifier type="weka.classifiers.functions.SMO"/>
30         </wrapper-classifier>
31       </query-strategy>

```

Figura 14 - Code Editor. Tema Claro.

Al final de la sección como se muestra en la Figura 15, tenemos tres botones:

- “Save” (Guardar): Salvar el archivo seleccionado en la pestaña activa del editor.
- “Load” (Cargar): Cargar un archivo desde su terminal en alguno de los formatos compatibles o la configuración de un experimento en formato XML en una nueva pestaña del editor.
- “Export” (Exportar): Exportar archivo de configuración de un experimento en formato XML.

<sup>6</sup> Por el momento XML, Java y JavaScript.

```

1  ?xml version="1.0" encoding="UTF-8" standalone="no"?>
2  <experiment>
3      <process evaluation-method-type="net.sf.jclal.evaluation.method.KFoldCrossValidation">
4          <rand-gen-factory seed="340.267" type="net.sf.jclal.util.random.JavaRandomFactory"/>
5          <file-dataset type="net.sf.jclal.dataset.WekaDataset">
6              <path>/JCLAL/datasets/ionosphere.arff</path>
7              <class-index>1</class-index>
8          </file-dataset>
9          <sampling-method type="net.sf.jclal.sampling.supervised.Resample">
10             <percentage-to-select>5.0</percentage-to-select>
11             <no-replacement>true</no-replacement>
12             <invert-selection>false</invert-selection>
13         </sampling-method>
14         <algorithm type="net.sf.jclal.activelearning.algorithm.ClassicalALAlgorithm">
15             <listener type="net.sf.jclal.listener.ClassicalReporterListener">
16                 <report-frequency>1</report-frequency>
17                 <report-on-file>true</report-on-file>
18                 <report-on-console>false</report-on-console>
19                 <report-directory></report-directory>
20                 <report-title>ReportKFold170044295</report-title>
21             </listener>
22             <scenario type="net.sf.jclal.activelearning.scenario.StreamBasedSelectiveSamplingScenario">
23                 <batch-mode type="net.sf.jclal.activelearning.batchmode.QBestBatchMode">
24                     <batch-size>1</batch-size>
25                 </batch-mode>
26                 <query-strategy type="net.sf.jclal.activelearning.singlelabel.querystrategy.EntropySamplingQueryStrategy">
27                     <maximal>true</maximal>
28                     <wrapper-classifier type="net.sf.jclal.classifier.WekaClassifier">
29                         <classifier type="weka.classifiers.functions.SMO"/>
30                     </wrapper-classifier>
31                 </query-strategy>

```

SAVE    LOAD    EXPORT

Figura 15 - Code Editor. Tema oscuro y Botones.

### 3.5 Experiments

Luego de haber realizado un experimento los resultados del mismo serán mostrados en la vista "Experiments" (Experimentos).

En primer lugar encontrará un grupo de tarjetas mostrando la información más relevante de los resultados obtenidos como: El total de instancias clasificadas, el nombre del dataset, el numero de instancias clasificadas correcta e incorrectamente, el porciento que esto representa del total y el tiempo que tomo la ejecución del experimento y la fecha de finalización del mismo. Más abajo encontrará una tarjeta donde podrá seleccionar entre varias graficas que mostrarán de manera grafica e interactiva, distintos datos del resultado del experimento. Ver Figura 16.

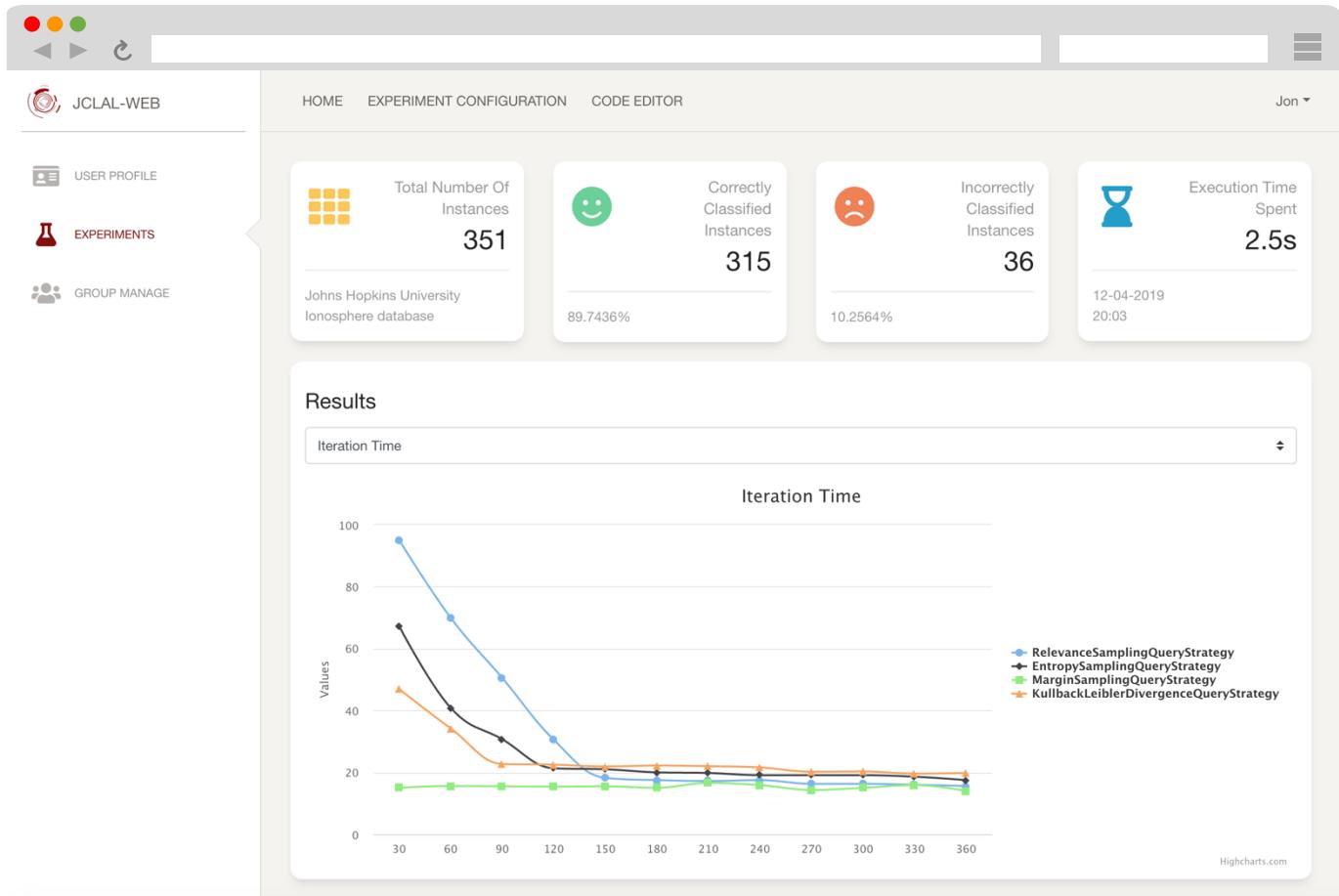


Figura 16 - Experiments. Resultados.

Luego encontraremos a la derecha una tarjeta llamada "Experiment Report" donde encontrara un reporte escrito de los resultados del experimento que podrá exportar en formato PDF o en texto plano mediante el botón "Export". A la derecha en otra sección llamada "Experiments List" podrá ver una lista con el historial de experimentación que ha realizado donde se mostrara dividido por columnas, el nombre, si ha sido realizado en equipo o solitario y la fecha de finalización de los mismos. Pulsando sobre los títulos de éstas columnas ("Name", "Team" y "Date") podrá filtrar dicha lista. Ver Figura 17.

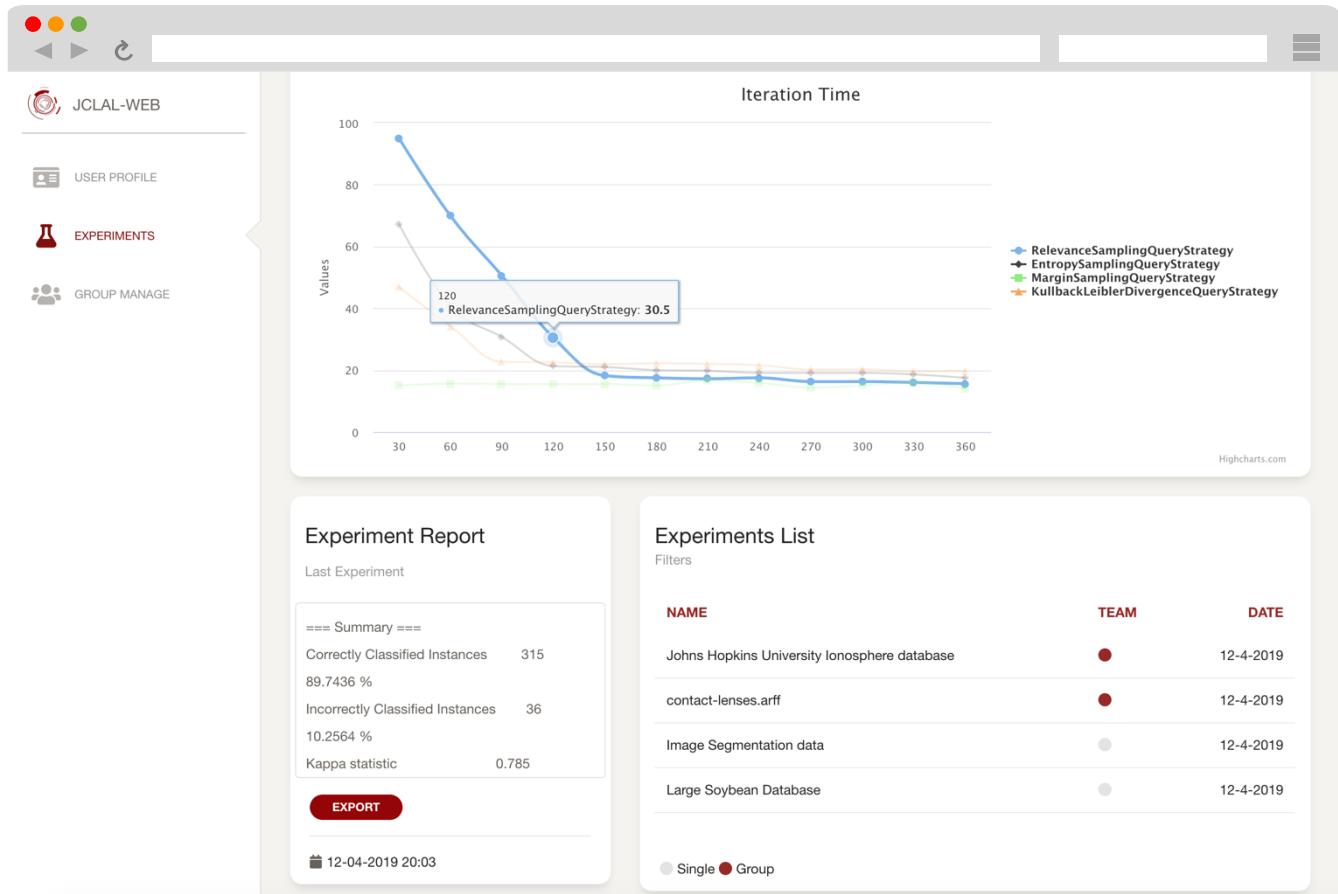


Figura 17 - Experiments. Historial y Reporte.

## 4 Casos de estudio

Para un mejor entendimiento de la aplicación web JCLAL-WEB se presentará un caso de estudio que muestra cómo se configuran y ejecutan los experimentos en JCLAL-WEB.

### 4.1 Caso de estudio

De forma general cada experimento debe definir el conjunto de datos a utilizar, luego se instancian y configuran los elementos que componen el AL.

Tabla 1 - Caso de Estudio.

Dataset	Johns Hopkins University Ionosphere database: Compuesto por 351 ejemplos
Método de evaluación	kFoldCrossValidation 10 folds.
Algoritmo	ClassicalALAlgorithm Máximo número de iteraciones: 100. Conjunto de ejemplos etiquetados: 5% del conjunto de entrenamiento.
Escenario	StreamBasedSelectiveSamplingScenario Batch: QBestBatchMode Ejemplos añadidos por cada iteración: 1
Estrategia de consulta	MarginSamplingQueryStrategy
Clasificador base	SMO

A continuación se muestra la configuración del experimento y posterior ejecución de este utilizando JCLAL-WEB:

Se selecciona la opción de nueva configuración en la vista principal del sistema como se aprecia en la Figura 18. En la interfaz de configuración del experimento seleccionar los parámetros deseados, en la sección correspondiente al conjunto de datos (Dataset) se escoge el dataset, así como se configura el resto de los parámetros como el tipo de dataset y el índice de la clase. Figura 19 y 20.



Figura 19 - Accediendo a "Experiment Configuration".

Figura 18 – Configurando experimento. Proceso..

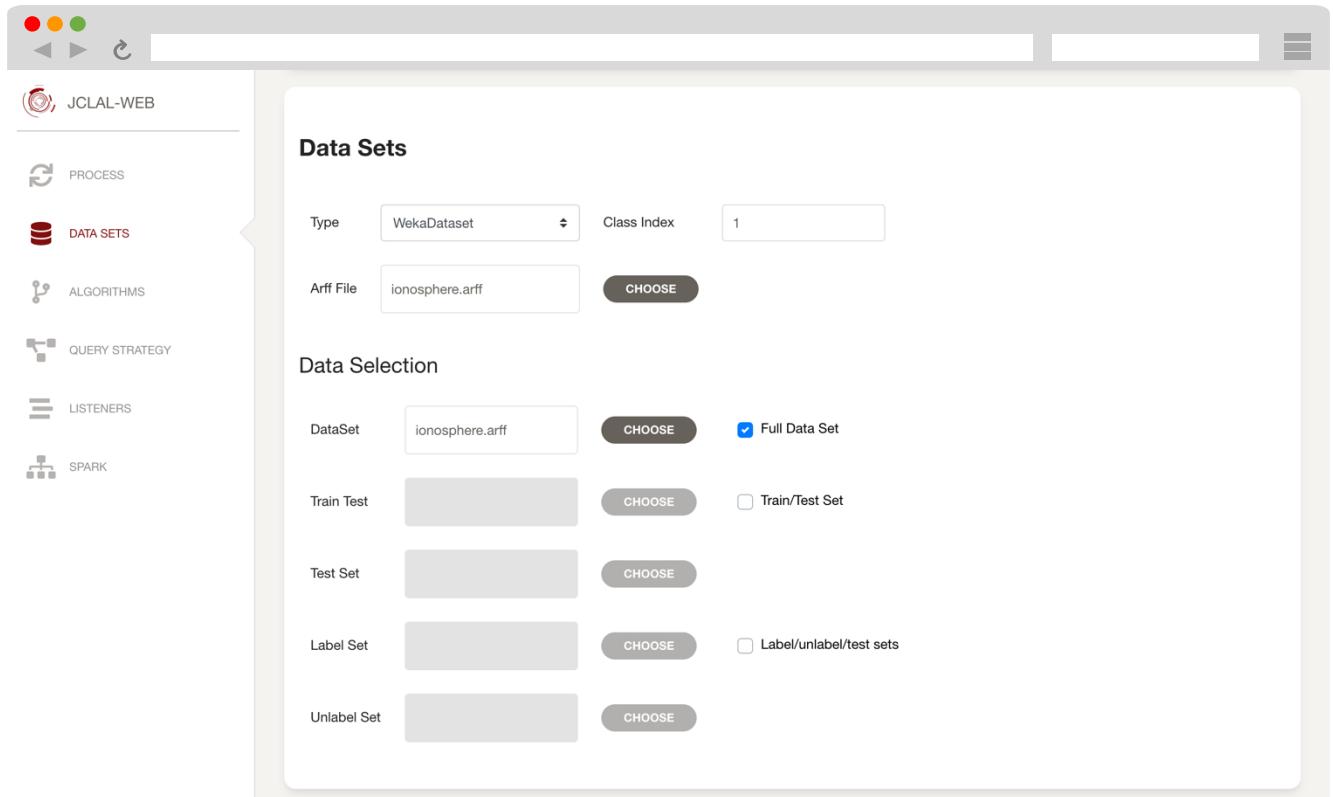


Figura 20 - Configurando experimento. Conjunto de datos.

En la sección de Algoritmo que se puede apreciar en la Figura 21, seleccionamos el algoritmo clásico de AL y el número máximo de iteraciones, en este caso 100. En esta pestaña también seleccionamos el escenario **PoolBasedSampling** y el **batch** de ejemplos a usar en este escenario.

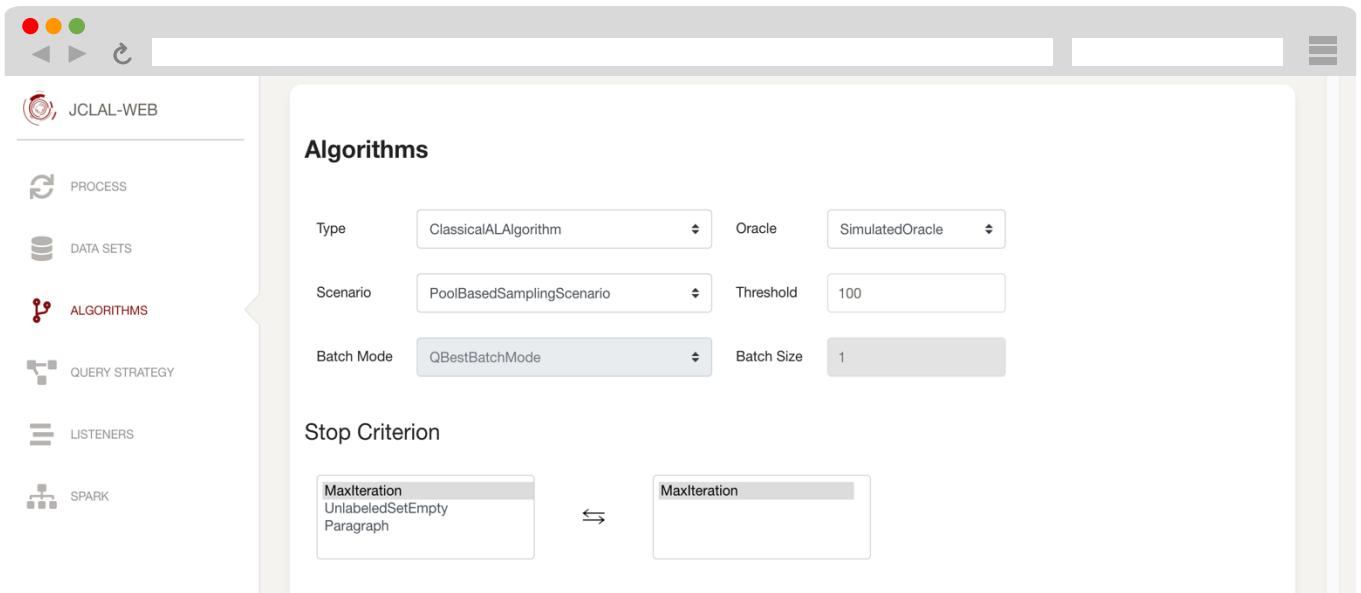


Figura 21 - Configurando experimento. Algoritmo.

Por último en la sección de la Estrategia de consulta seleccionamos **MarginSamplingQueryStrategy** y como clasificador base **SMO** (ver Figura 22).

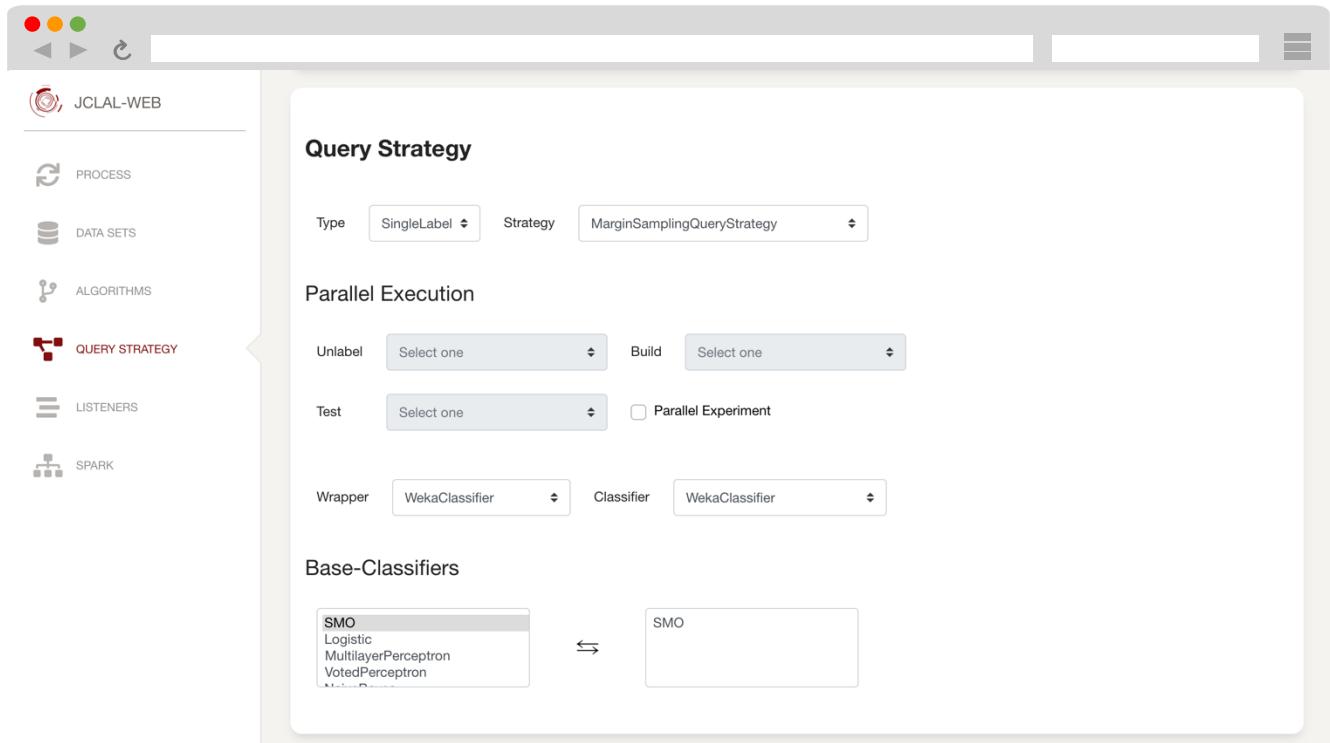


Figura 22 - Configurar experimento. Estrategia de consulta.

Una vez realizados estos pasos se puede pulsar el botón de ejecutar el experimento.

## Referencias bibliográficas

- [1] B. Settles, "Active learning literature survey," *Univ. Wisconsin, Madison*, vol. 15, no. 2, pp. 201–221, 2010.
- [2] B. Settles, "Active learning literature survey," *Univ. Wisconsin, Madison*, vol. 15, no. 2, pp. 201–221, 2010.
- [3] O. Reyes, E. Pérez, M. Del, C. Rodríguez-Hernández, H. M. Fardoun, and S. Ventura, "JCLAL: A Java Framework for Active Learning," *J. Mach. Learn. Res.*, vol. 17, pp. 1–5, 2016.
- [4] L. M. Sánchez Velázquez, "VISUALJCLAL : ENTORNO DE USUARIO PARA EL FRAMEWORK JCLAL," Universidad de Holguín "Oscar Lucero Moya," 2017.
- [5] S. Luján-Mora, "Programación de aplicaciones web: historia, principios básicos y clientes web," Oct. 2002.