

Universidades de Burgos, León y Valladolid

Máster universitario

Inteligencia de Negocio y Big Data en Entornos Seguros



Trabajo Fin de Máster

Plugin Knime para conexión con Moodle

Presentado por Francisco Gil Rodríguez
en la Universidad de Burgos — 7 de agosto de 2022

Tutores: Dr. José Francisco Díez Pastor y
Dr. César Ignacio García Osorio

Universidades de Burgos, León y Valladolid



Máster universitario en Inteligencia de Negocio y Big Data en Entornos Seguros

D. José Francisco Díez Pastor, profesor del departamento de Ingeniería Informática, área de Lenguajes y Sistemas Informáticos.

D. César Ignacio García Osorio, profesor del departamento de Ingeniería Informática, área de Lenguajes y Sistemas Informáticos.

Exponen:

Que el alumno D. Francisco Gil Rodríguez, con DNI 42183724E, ha realizado el Trabajo final de Máster en Inteligencia de Negocio y Big Data en Entornos Seguros titulado «Plugin Knime para conexión con Moodle».

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 7 de agosto de 2022

Vº. Bº. del Tutor:

Vº. Bº. del Tutor:

D. José Francisco Díez Pastor

D. César Ignacio García Osorio

Resumen

TODO: abstract castellano

Descriptores

TODO: descriptores castellano

Abstract

TODO: abstract inglés

Keywords

TODO: descriptores inglés

Índice general

Índice de figuras

Índice de tablas

Memoria

Introducción

Descripción del contenido del trabajo y del estructura de la memoria y del resto de materiales entregados.

Objetivos del proyecto

En este apartado se exponen los objetivos del proyecto, diferenciados entre objetivos generales, técnicos y personales.

2.1. Objetivos generales

- Desarrollar un plugin de Knime que permita importar datos desde una plataforma de teleformación Moodle, para su tratamiento posterior dentro de flujos de trabajo de Knime.
- Facilitar la incorporación en Knime de datos de cursos de cualquier plataforma Moodle, sin necesidad de realizar ajustes personalizados dentro de la plataforma de formación.
- Orientar la solución al perfil de Profesor, de forma que cualquier usuario con este perfil pueda realizar estudios desde Knime de los datos relativos a sus cursos.
- Implementar un ejemplo práctico utilizando el plugin de Knime implementado en un flujo de trabajo relacionado con Aprendizaje Supervisado.

2.2. Objetivos técnicos

- Desarrollar un plugin para Knime en lenguaje de programación Java.
- Explorar la arquitectura de Knime para elegir el tipo de componentes a desarrollar que mejor se adapten a la solución requerida.

- Conocer a fondo la metodología de programación en Knime mediante el estudio de la documentación existente y la inspección de plugins de Knime similares.

2.3. Objetivos personales

- Aplicar los conocimientos adquiridos dentro del Máster en al ámbito educativo y más específicamente en el de la formación online.
- Conocer más a fondo las herramientas que permiten el análisis de datos desde la interfaz de usuario, tanto a nivel de usuario como a nivel de desarrollador.
- Contribuir a la mejora de la formación online al facilitar que los profesores puedan realizar análisis sobre sus cursos sin conocimientos de programación.

Conceptos teóricos

Industrial Tools Every Data Scientist Should Know

3.1. Aprendizaje no supervisado

3.2. ETL

3.3. Arquitectura Knime

<https://datascience.fm/industrial-tools-every-data-scientist-should-know/>
<https://www.kdnuggets.com/2021/01/best-tool-data-blending-knime.html>

Automatic workflows

Sobre Nodos, extensiones, workflows Sobre knime hub

3.4. Moodle

Acceso a Logs de moodle

Técnicas y herramientas

4.1. Eclipse

4.2. Knime

4.3. Java

4.4. GitHub

4.5. Insomnia Rest Client

4.6. Docker

4.7. Moodle API

4.8. UBUMonitor

4.9. Otras herramientas utilizadas

- Visual Studio Code

- LaTeX

Anonymization (salt) <https://auth0.com/blog/adding-salt-to-hashing-a-better-way-to-store-passwords/>

Aspectos relevantes del desarrollo del proyecto

Este apartado pretende recoger los aspectos más interesantes del desarrollo del proyecto, comentados por los autores del mismo. Debe incluir desde la exposición del ciclo de vida utilizado, hasta los detalles de mayor relevancia de las fases de análisis, diseño e implementación. Se busca que no sea una mera operación de copiar y pegar diagramas y extractos del código fuente, sino que realmente se justifiquen los caminos de solución que se han tomado, especialmente aquellos que no sean triviales. Puede ser el lugar más adecuado para documentar los aspectos más interesantes del diseño y de la implementación, con un mayor hincapié en aspectos tales como el tipo de arquitectura elegido, los índices de las tablas de la base de datos, normalización y desnormalización, distribución en ficheros³, reglas de negocio dentro de las bases de datos (EDVHV GH GDWRV DFWLYDV), aspectos de desarrollo relacionados con el WWW... Este apartado, debe convertirse en el resumen de la experiencia práctica del proyecto, y por sí mismo justifica que la memoria se convierta en un documento útil, fuente de referencia para los autores, los tutores y futuros alumnos.

5.1. Análisis

En la fase de análisis nos tenemos que asegurar de que se podrán cumplir los objetivos del proyecto:

- Acceso a Moodle para extracción de datos - Implementación y ejecución de nodos de forma aislada y en combinación con otros nodos de Knime, dentro de un workflow.

Fuente de datos real

- Explicar opciones de extracción de datos de Moodle. - Acotar el alcance de información a extraer de Moodle

Cursos de prueba

Posibilidad de incorporar fuentes de datos reales (asignatura)

Estudio de nodo de prueba y otros nodos similares (ejemplo paquete Twitter)

Programación en JAVA.

Estudio de métodos de acceso a los datos de Moodle. En este punto se detecta que los servicios web de Moodle no ofrecen acceso a todos los datos que se pueden visualizar desde la APP de Moodle. Aquí se puede hablar también de APP de Moodle y de acceso por Web Scraping.

5.2. Diseño

- Opciones de reutilización de código: UBUMonitor APP de Moodle
- Análisis mediante Insomnia de los datos obtenidos a través de la API
- Organizar los sprints para abordar el proyecto de forma organizada
- Diseño de nodos.

Se descubre que los nodos no se pueden ejecutar a nivel de código. Esto es, no podemos inyectar la funcionalidad completa de un nodo dentro de otro, ejecutándolo internamente. Knime ha impuesto esta restricción de diseño para evitar la alta dependencia entre nodos y los problemas que ello conllevaría a nivel de mantenimiento cuando un nodo queda obsoleto.

5.3. Implementación

5.3.1. Acceso a datos en Moodle

Comentar aquí solución web services, limitaciones tanto de acceso a datos como de

Finalmente se opta por web scraping utilizando la cuenta del profesor para acceder a los logs.

Login con webservice de la APP de Moodle. Esto implica que la implementación de Moodle debe tener activa la opción de acceso de la APP.

5.3.2. Desarrollo de nodos en Knime

Entorno de desarrollo:

Cómo montar el entorno de desarrollo según Knime Nodo de muestra
Otros recursos (código de nodos desarrollados) Añadir librerías (dependencias en proyecto knime)

Extensión:

- KNIME Moodle Integration

Nodos desarrollados:

- Moodle Connector.

Establece la conexión con una plataforma Moodle. Requiere una cuenta con perfil de profesor. Requiere que la plataforma Moodle tenga activado el acceso a la aplicación Móvil.

Output Ports.

0. A connection that can be used to access Moodle's API.

URL usuario contraseña

Salida: token

- Moodle Courses.

core_course_get_courses core_course_get_courses_by_field core_course_search_courses

Input Ports:

0 Connection to Moodle's API (token) 1 Table containing...

Output Ports:

0 Información de cursos (ID cursos)

- Moodle Users.

core_enrol_get_enrolled_users anonymization

- Moodle Reports

logs Live logs Activity Reports Overview Statistics Course Participation
Activity completion Statistics

- Reutilización de nodos desde programación Explicar la ventaja de poder utilizar nodos desde programación. Los nodos son clases que se pueden instanciar. La ventaja es que están pensados para trabajar con

elementos (clases) de Knime. Así que esta estrategia será interesante solo cuando vayamos a trabajar con elementos de Knime, como conversiones de tipos desde o hacia clases Knime.

Explicar por qué no se ha podido utilizar esta estrategia.

<https://forum.knime.com/t/using-node-without-gui/2044/6>

Ejemplos de reutilización:

- JSON to table

Workflows desarrollados:

Nota: los workflows compartidos son casi tan importantes como los nodos desarrollados, ya que permiten el uso directo de soluciones

- Workflow básico (extraer logs) - Workflow clasificación - Workflow clustering

- Dependencias con proyectos externos

UBUMonitor Versión 2.10.2 20220426 <https://github.com/yjx0003/UBUMonitor/releases>

JSON In Java (20220320) <https://github.com/stleary/JSON-java>

org.knime.json Incluido en: https://update.knime.org/analytics-platform/UpdateSite_latest

¿ Moodle-REST-API-for-Java ? En principio parece que está más mantenido el UBUMonitor <https://github.com/yjx0003/Moodle-REST-API-for-Java>

5.4. Workflows

Nodos externos utilizados en Workflows

<https://hub.knime.com/redfield/extensions/se.redfield.arx.feature/latest>

Trabajos relacionados

6.1. UBU Monitor

Conclusiones y Líneas de trabajo futuras

Todo proyecto debe incluir las conclusiones que se derivan de su desarrollo. Éstas pueden ser de diferente índole, dependiendo de la tipología del proyecto, pero normalmente van a estar presentes un conjunto de conclusiones relacionadas con los resultados del proyecto y un conjunto de conclusiones técnicas. Además, resulta muy útil realizar un informe crítico indicando cómo se puede mejorar el proyecto, o cómo se puede continuar trabajando en la línea del proyecto realizado.

Integración con Moodle a través de web services. Permitir una integración que permita añadir estructuras más complejas... como analizar un grupo de cursos. Esta línea requiere no pensar solo en el rol de profesor, sino en la intervención por parte de la institución para facilitar la integración y el acceso a los datos

Colección de Flujos de trabajo típicos con Knime. Knime permite crear flujos de trabajo y compartirlos, de forma que sea fácil su reutilización.

Knime en cloud o servidor para flujos de trabajo "permanentes"(cron). Lo que hemos visto se basa solo en ejecución de knime desde local. Extender a flujos

Comunicación bidireccional con Moodle. Tomar acciones a partir de los resultados del procesamiento de los datos. Por ejemplo, enviar un correo a los alumnos clasificados como en riesgo de abandono.

Apéndice

Apéndice A

Plan de Proyecto Software

A.1. Introducción

A.2. Planificación temporal

A.3. Estudio de viabilidad

A.3.1. Viabilidad económica

A.3.2. Viabilidad legal

Apéndice B

Especificación de Requisitos

- B.1. Introducción
- B.2. Objetivos generales
- B.3. Catalogo de requisitos
- B.4. Especificación de requisitos

Apéndice C

Especificación de diseño

- C.1. Introducción
- C.2. Diseño de datos
- C.3. Diseño procedimental
- C.4. Diseño arquitectónico

Apéndice D

Documentación técnica de programación

- D.1. Introducción
- D.2. Estructura de directorios
- D.3. Manual del programador
- D.4. Compilación, instalación y ejecución del proyecto
- D.5. Pruebas del sistema

Apéndice E

Documentación de usuario

- E.1. Introducción
- E.2. Requisitos de usuarios
- E.3. Instalación
- E.4. Manual del usuario