Algoritmos y Estructura de Datos II

Trabajo Práctico Integrador – Primer Cuatrimestre de 2023

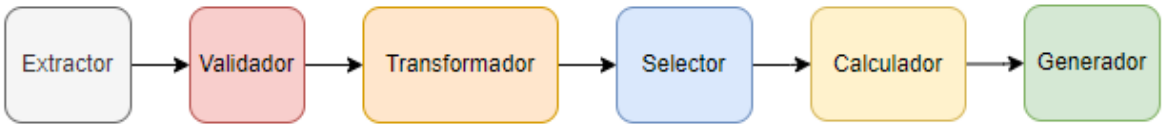
Requisitos de aprobación del trabajo

- Entrega de código fuente sin errores o excepciones no capturadas
- Presentación del trabajo y verificación del cumplimiento de los requerimientos solicitados.

Requerimientos mínimos del modelo

Extensibilidad: Es necesario que el modelo pueda extenderse a nuevos tipos de nodos de procesamiento, calculadores, extractores, generadores, transformadores, etc.
Mantenibilidad: Hacer uso de los principios de POO para construir un producto que sea de fácil mantenimiento, por ejemplo evitando la duplicación de código y desacoplando los subsistemas cuando sea posible.
Gestión de errores: El sistema debe capturar todos los errores posibles en tiempo de ejecución y clasificarlos según corresponda con una jerarquía de excepciones.

El pipeline de datos



El objetivo de este trabajo es implementar un modelo basado en el paradigma de la programación orientada a objetos que permita construir y ejecutar un pipeline de datos con las condiciones y enunciados propuestos.

El pipeline

El sistema permite construir un pipeline de datos compuesto por diferentes componentes o módulos. Este pipeline generado funciona como una secuencia de pasos a realizar donde el inicio debe ser con mínimo un módulo de extracción o carga de datos, y el final debe ser con un mínimo de un módulo de generador de datos. Puede existir más de un punto de entrada en el pipeline, es decir, puede haber más de un módulo extractor, como así también pueden existir más de un punto de salida (pueden haber varios módulos generadores que produzcan distintos datasets). La secuencia dentro de un pipeline no es necesariamente lineal ya que existen módulos (de tipo Selector) que pueden producir bifurcaciones, como así también módulos (de tipo Computación múltiple) que permiten ejecutar en varias ramas en paralelo.

Construcción y ejecución

Cuando se construye el pipeline se van agregando módulos de cierto tipo para adecuar el diseño deseado (ver abajo diseño a presentar). Una vez construido, el pipeline se ejecuta para producir el

resultado esperado según la entrada. La ejecución de un pipeline puede detenerse arrojando un error si alguno de los módulos falla en su ejecución.

Módulos o Componentes

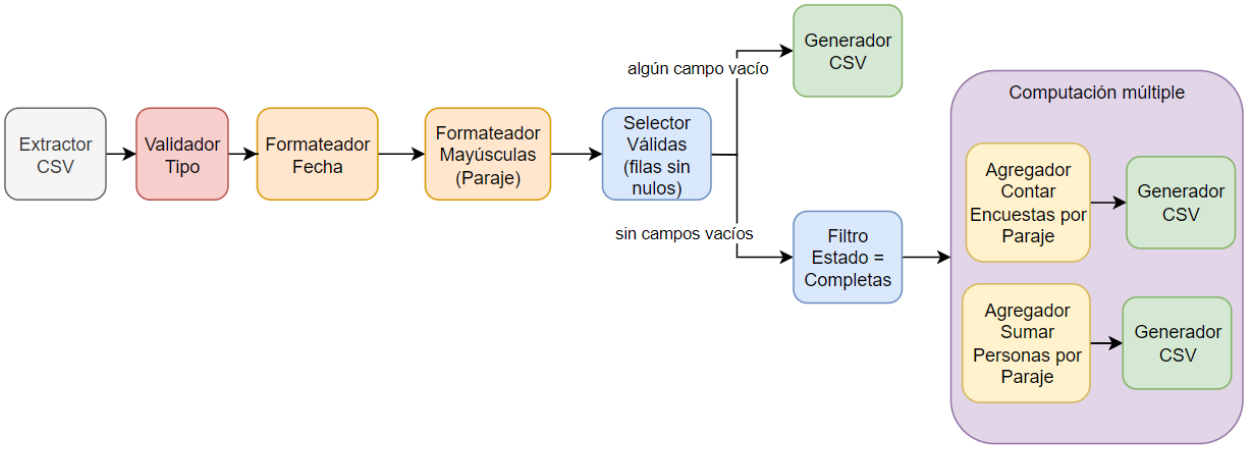
Cada módulo o componente de un pipeline recibe una entrada de datos, generalmente en forma de matriz/tabla, realiza alguna acción que depende del tipo de componente y entrega como salida de su ejecución otra matriz/tabla con los datos procesados. Algunos módulos pueden entregar más de una salida de datos, ya que pueden bifurcar el flujo. También algunos módulos pueden recibir información adicional de entrada para utilizar como configuración de ese módulo de forma dinámica (por ejemplo, un formateador de fecha puede recibir los datos y como adicional la columna que contiene las fechas a formatear).

Definiremos los siguientes tipos de componentes posibles dentro de un pipeline, pero se debe considerar que eventualmente existan otros:

Tipo de Módulo/Componente	**Descripción**
Extractor	Permite extraer datos de alguna fuente, por ejemplo archivo CSV.
Validador Tipo	Se puede instanciar con cierta configuración para realizar la validación de tipos de datos de cada campo (columna). Por ejemplo, un validador de 3 columnas de tipo String. Si da error, falla el pipeline.
Formateador	Permite cambiar el formato de un campo/columna.
Selector	Permite seleccionar subconjuntos de filas o columnas para entregar más de una salida y así bifurcar el flujo.
Filtro	Es un subtipo de Selector que permite seleccionar un subconjunto de filas según cierto criterio y entrega una salida.
Agregador	Genera una agregación de la entrada, agrupando por cierto campo/columna computa una métrica, por ejemplo contar, sumar, promediar, etc.
Computación múltiple	Es un módulo que contiene otros pipelines, entrega la misma entrada que recibe como entrada a cada uno de sus pipelines.
Generador	Produce una fuente de datos según el tipo de generador, por ejemplo un Generador CSV producirá un archivo de valores separados por coma (CSV).

Escenario de prueba

Este escenario deberá ser preparado para presentar la fecha de entrega. Se entrega también un dataset en formato CSV que será el punto de entrada del pipeline que contiene datos de encuestas realizadas en distintos parajes, cada encuesta corresponde a una fila. El pipeline a construir para este caso es el siguiente:



El módulo Validador debe validar lo siguiente:

- Columna id: números enteros positivos
- Columna fecha_inicial: fecha válida
- Columna estado_encuesta: debe ser alguno de estos valores
    - Vacía (‘’)
    - Completa
    - Imposibilitada
    - De Prueba
    - En proceso
- Columna cantidad_personas: números enteros positivos

El módulo formateador debe transformar la fecha_inicial con el formato: DD/MM/AAAA

El módulo Selector Válidas debe separar la entrada y devolver 2 salidas:

- Encuestas (filas) que tengan algún campo vacío (‘’)
- Encuestas que no tienen campos vacíos

El módulo Filtro debe seleccionar sólo aquellas encuestas que tengan como estado_encuesta = Completa.

Se deberán generar al finalizar la ejecución 3 archivos CSV:

- Encuestas con algún campo vacío (nulo). Columnas requeridas: las mismas que el archivo de entrada.
- Cantidad de encuestas en cada paraje. Columnas requeridas: paraje, cantidad_encuestas.
- Cantidad de personas en cada paraje. Columnas requeridas: paraje, cantidad_personas.