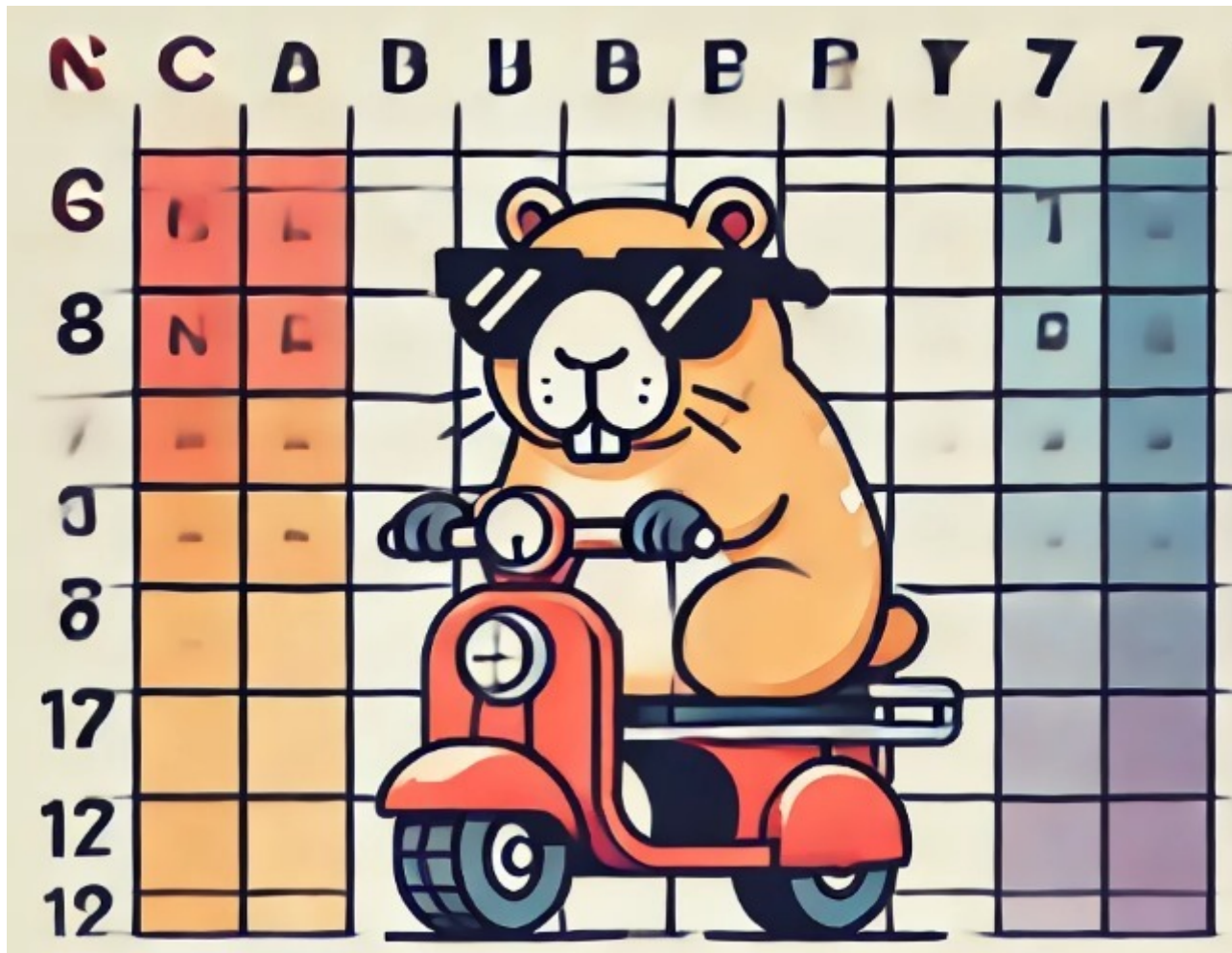


Índice

Hola soy Kapibara, soy un animal muy amigo de los pandas, te invito a revisar mi documentación para que me puedas conocer mejor 😊

Documentación Web generada con JavaDocs



- Proyecto: **Librería de Java**
 - 1. Objetivo y Alcance
- Objetivo del Proyecto
- Alcance del Proyecto
 - 2. Descripción de Alto Nivel del Sistema
 - 3. Requerimientos Funcionales
 - 4. Requerimientos No Funcionales
 - 5. Estructura del repositorio
 - Desarrollado por:

Proyecto: **Librería de Java**

1. Objetivo y Alcance

Objetivo del Proyecto

El proyecto tiene como objetivo desarrollar una **librería en Java** llamada KAPIBARA que permita manipular y analizar datos en forma tabular (2 dimensiones). La librería debe proporcionar estructuras de datos y operaciones que faciliten la manipulación de datos tabulares, como lectura, modificación y análisis de los mismos, sin depender de librerías externas.

Además, el proyecto contempla la posibilidad de **futuras extensiones** y debe minimizar el impacto de futuras modificaciones en la arquitectura del sistema. Aunque no se hará foco en la eficiencia de las operaciones inicialmente, se considerará la inclusión de mecanismos para medir el **costo temporal** de la ejecución de las mismas.

Alcance del Proyecto

El proyecto abarca los siguientes aspectos:

1. **Implementación de estructuras de datos:** Que permitan almacenar y manipular datos en dos dimensiones.
2. **Operaciones sobre las estructuras de datos:** agregar, eliminar y modificar celdas, filtrado, y operaciones básicas de análisis.
3. **Extensibilidad:** Se deberá diseñar una arquitectura que permita agregar nuevas funcionalidades en el futuro con mínimo impacto.
4. **Medición de costo temporal:** Incorporación de una función que permita medir el tiempo de ejecución de las operaciones más relevantes.

2. Descripción de Alto Nivel del Sistema

La librería consiste en una biblioteca de Java para poder manipular estructuras tabulares de 2 dimensiones, el usuario podrá importar esta librería en su propio proyecto y acceder a los métodos implementados para poder leer, agregar, modificar, filtrar y ordenar los datos provistos por el usuario

3. Requerimientos Funcionales

Los principales requerimientos funcionales del sistema incluyen:

- **Macro-requerimiento 1: Implementación de estructuras de datos**
 - RF 1.1: TAD de dataframe de 2 dimensiones con n filas y m columnas,
 - RF 1.2: TAD para una fila
 - RF 1.3: TAD para una columna
 - RF 1.4: Implementar etiquetas(labels) para poder indexar filas y columnas
- **Macro-requerimiento 2: Operaciones sobre las estructuras de datos**
 - RF 2.1: Ingresar celdas null o NA
 - RF 2.2: Modificar las celdas null
 - RF 2.3: Cambiar el dato de una celda
 - RF 2.4: Agregar y eliminar filas y columnas de una tabla

- RF 2.5: Mostrar la cantidad de filas y columnas de la tabla
 - RF 2.6: Mostrar los tipos de datos de cada columna
 - RF 2.7: Mostrar cuales son las etiquetas de las filas y las columnas
 - RF 2.8: Cargar desde el disco una tabla en formato csv
 - RF 2.9: Definir el carácter delimitador de columnas y si se utilizarán las etiquetas
 - RF 2.10: Visualizar la tabla en una forma comprensible
 - RF 2.11: Definir un máximo de columnas y filas cuando se quieran visualizar
 - RF 2.12 Debe rechazar archivos con formatos incorrectos y proporcionar mensajes de error claros.
 - RF 2.13 Filtrar por columna y valor de celda
 - RF 2.14 Generar una tabla nueva a partir de una copia profunda de otra previamente creada.
 - RF 2.15 Generar una tabla nueva a partir de una estructura de 2 dimensiones nativa de Java.
 - RF 2.16 Mostrar las n primeras o n últimas filas
 - RF 2.17 Generar una copia profunda de una tabla
- **Macro-requerimiento 3: Extensibilidad**
 - RF 3.1: Que pueda permitir agregar nuevos tipos de datos para las celdas
 - RF 3.2: Que se pueda cambiar el formato de la visualización de las tablas
- **Macro-requerimiento 4: Medición de costo temporal**
 - RF 4.1: Mostrar en pantalla cuanto tarda el programa en cargar un csv

4. Requerimientos No Funcionales

Los requerimientos no funcionales definen cómo debe comportarse el sistema en términos de rendimiento, seguridad y usabilidad.

- **RNF 1:** La librería debe poder ser usada en Java 8 o superior
- **RNF 2:** El procesamiento de archivos de hasta 10,000 filas debe completarse en menos de 1 segundo.
- **RNF 3:** Seguridad, permitir seleccionar columnas privadas para que no sean accesibles para el usuario
- **RNF 4:** Rendimiento, benchmarking de métodos

5. Estructura del repositorio

- **src/kapibara:** Contiene el código fuente principal de la aplicación
- **src/tests/:** Incluye las pruebas para asegurar la funcionalidad correcta del código.
- **src/enums/:** Enums.
- **temp/:** Archivos temporales, por ejemplo cuando se guarda una tabla en los runs de prueba.
- **data/:** Datasets de prueba.
- **docs/:** Documentación adicional del proyecto, como guías de usuario, referencias de API y diagramas.

Desarrollado por:

- [AgustinRebechi](#)
- [aecheverri](#)
- [dabaron13](#)
- [rriveraro](#)

- [cvidalsastre](#)