

Facultad de Ingeniería Estadística e Informática

Escuela Profesional de Ingeniería Estadística e Informática



Lenguajes de Programación I

Evaluación del Código Fuente

Profesor: Fred Torres Cruz

Estudiante: Roberth Carlos Gonzales Mauricio

Fecha: 20 de junio de 2024

Semestre: 3ro Grupo: B

Introducción a la Evaluación del Código Fuente

Se realizó la evaluación del código fuente de un archivo utilizando las métricas de Lines of Code (LOC). Estas métricas nos permiten evaluar la calidad y mantenibilidad del código.

Evaluación del Código Fuente

Para evaluar el tamaño y la calidad del código fuente desarrollado, hemos aplicado métricas de Lines of Code (LOC). A continuación, presentamos una tabla con los resultados obtenidos del archivo de código evaluado.

Cuadro 1: Evaluación de métricas LOC del archivo de código. La fórmula para calcular la Comment to Code Ratio (CCR) se define como el cociente entre las líneas de comentario (CLOC) y las líneas ejecutables de código (ELOC), proporcionando una medida de la proporción de comentarios respecto al código ejecutable.

Las métricas LOC nos proporcionan información valiosa sobre el tamaño y la estructura del código. El número total de líneas de código (LOC) es 63, mientras que las líneas ejecutables (ELOC) son 53. Es notable que no hay líneas de comentarios (CLOC), lo que resulta en una relación de comentarios por línea de código (CCR) de 0. Esto sugiere que el código puede carecer de documentación interna, lo cual puede dificultar su mantenibilidad y comprensión por parte de otros desarrolladores.

Además, el número de líneas en blanco (BLOC) es 10, lo cual indica que hay una cantidad mínima de espacios en blanco, posiblemente afectando la legibilidad.

Métricas de Código Fuente

Lines of Code (LOC): 63

Executable Lines of Code (ELOC): 53

Comment Lines of Code (CLOC): 0

Comment to Code Ratio (CCR): 0.00

Non-Comment Lines of Code (NCLOC): 63

Blank Lines of Code (BLOC): 10

Código Fuente

```
import streamlit as st

import pandas as pd

import plotly.express as px

from datetime import datetime


def calculate_age(born):

    today = datetime.today()

    return today.year - born.year


class ManejadorCSV:

    def __init__(self, nombre_archivo):

        self.nombre_archivo = nombre_archivo

        self.df = pd.read_csv(nombre_archivo)

        self.df['Fecha de nacimiento'] = pd.to_datetime(self.df['Fecha de
nacimiento'])
```

```

def mostrar_datos(self):

        self.df.insert(3, 'Edad', self.df['Fecha de
nacimimiento'].apply(calculate_age))

        st.write(self.df)

        x_axis = st.selectbox("Selecciona la columna para el eje X",
options=self.df.columns)

        y_axis = st.selectbox("Selecciona la columna para el eje Y",
options=self.df.columns)

        fig = px.scatter(self.df, x=x_axis, y=y_axis)

        st.plotly_chart(fig)


def guardar_como(self, nuevo_nombre_archivo):

    self.df.to_csv(nuevo_nombre_archivo, index=False)

    st.success("Datos escritos en {}".format(nuevo_nombre_archivo))


def preview_data(self):

    st.subheader("Vista previa de datos:")

    st.write(self.df.head(10))


def calculate_statistics(self):

    st.subheader("Estadísticas de las edades:")

    age_column = 'Fecha de nacimiento'

    if age_column in self.df.columns:

```

```

        self.df['Edad'] = self.df[age_column].apply(calculate_age)

        mean_age = self.df['Edad'].mean()

        median_age = self.df['Edad'].median()

        std_age = self.df['Edad'].std()

        st.write(f"Media de edades: {mean_age}")

        st.write(f"Mediana de edades: {median_age}")

        st.write(f"Desviación estándar de edades: {std_age}")


def main():

    st.title("CSV ESTADISTICA")

    nombre_archivo = st.file_uploader("CARGUE UN ARCHIVO CSV",
type=['csv'])

    if nombre_archivo is not None:

        manejador = ManejadorCSV(nombre_archivo)

        st.subheader("Datos del archivo original:")

        manejador.mostrar_datos()

        nuevo_nombre_archivo = st.text_input("Guardar como (nombre del
archivo CSV).csv")

        if st.button("Guardar"):

            if nuevo_nombre_archivo:

                manejador.guardar_como(nuevo_nombre_archivo)

        if st.checkbox("Vista previa de datos"):

            manejador.preview_data()

```

```
        if st.checkbox("Calcular estadísticas de edades"):

            manejador.calculate_statistics()


if __name__ == "__main__":

    main()
```