```
import streamlit as st
import pandas as pd
import sqlite3

def load_data(file):
    # Load the file into a DataFrame
    if file.name.endswith('.csv'):
        return pd.read_csv(file)
    else:
        st.error("Unsupported file format. Please upload a CSV file.")

# Streamlit UI
st.title(":rainbow[DBMS Operations Web App]")

# Upload a CSV file
uploaded_file = st.file_uploader(":green[Choose a CSV file]", type="csv")
if uploaded_file:
    df = load_data(uploaded_file)
    st.write(":gray[Data Preview:]", df.head())

    # Store DataFrame in SQLite to support SQL queries
    conn = sqlite3.connect(':memory:')
    df.to_sql('uploaded_data', conn, index=False, if_exists="replace")

    # Select Operation ---
    st.subheader(":red[Select Operation]",divider='green')
    condition = st.text_input("Enter selection condition (e.g., age > 30)")
    if condition:
        query = f"SELECT * FROM uploaded_data WHERE {condition}"
        try:
            result = pd.read_sql(query, conn)
```

```
        st.write("Selection Result:", result)
    except Exception as e:
        st.error(f"Error: {e}")

# Projection Operation ---
st.subheader(":blue[Projection Operation]",divider='violet')
columns = st.multiselect("Select columns to display", df.columns)
if columns:
    query = f"SELECT {', '.join(columns)} FROM uploaded_data"
    result = pd.read_sql(query, conn)
    st.write("Projection Result:", result)

# Aggregation Part ---
st.subheader(":orange[Aggregation]",divider='green')
agg_column = st.selectbox("Select column for aggregation", df.columns)
agg_func = st.selectbox("Choose aggregation function", ["SUM", "AVG", "COUNT"])
if agg_column and agg_func:
    query = f"SELECT {agg_func}({agg_column}) FROM uploaded_data"
    result = pd.read_sql(query, conn)
    st.write(":orange[Aggregation Result:]", result)

conn.close()
```