

```

import streamlit as st
import xml.etree.ElementTree as ET
import re
from bs4 import BeautifulSoup

# Define text cleaning functions
def strip_html(text):
    """Removes HTML tags from text."""
    soup = BeautifulSoup(text, "html.parser")
    return soup.get_text()

def remove_between_square_brackets(text):
    """Removes text enclosed within square brackets."""
    return re.sub(r'\[[^\]]*\]', '', text)

def denoise_text(text):
    """Combines all cleaning functions to clean the text."""
    text = strip_html(text)
    text = remove_between_square_brackets(text)
    text = re.sub(' ', ' ', text)
    return text

# XML Parsing function
def parse_xml_file(file):
    """Parses an XML file and extracts content as a string."""
    tree = ET.parse(file)
    root = tree.getroot()
    xml_content = ET.tostring(root, encoding='utf8').decode('utf8')
    return xml_content

# Streamlit Interface
st.title("XML Text Cleaner")

st.write("Upload an XML file to parse and clean the text data.")

# File uploader widget
uploaded_file = st.file_uploader("Choose an XML file", type="xml")

# Check if a file has been uploaded
if uploaded_file is not None:
    # Parse and clean XML content
    raw_xml = parse_xml_file(uploaded_file)
    cleaned_text = denoise_text(raw_xml)

    # Display the results
    st.subheader("Raw XML Content")
    st.text_area("This is the raw XML content:", raw_xml, height=300)

    st.subheader("Cleaned Text")
    st.text_area("This is the cleaned text:", cleaned_text, height=300)
else:
    st.write("Please upload an XML file to see the cleaned output.")

```