Lesson Plan: Build a Moodio App

Age Group: 12–16
Duration: 60 minutes
Topic: Introduction to Text Classification with Moodio

Goals

- Understand the concept of **text mood / sentiment**.
- Learn how computers **analyze sentences** and map them to moods.
- Build and explore a **simple interactive web app** using Python and Streamlit.
- Practice safe coding for **kid-friendly content**.

Topics Introduced

1. **Text Classification / Sentiment Analysis**
    - Positive, Neutral, Negative text polarity.
    - Using simple NLP library (**TextBlob**).
2. **Web App Basics**
    - Streamlit input/output widgets.
    - Displaying emoji and messages.
3. **Kid-Safe Filtering**
    - Filtering inappropriate words.
    - Ensuring safe and age-appropriate content.
4. **Logic Flow / Debugging**
    - Input → Filter → Analysis → Output.
    - Understanding program flow.

Activity Explanation

1. **Introduction (5 min)**
    - Discuss how words express emotions.
    - Show example sentences and ask students to guess the mood.
2. **Demo App (10 min)**
    - Enter sample sentences in Moodio.
    - Show emoji and explanation.
    - Enable Teacher Mode to illustrate the logic flow.
3. **Hands-On Exercise (25 min)**
    - Students type their own sentences.
    - Predict the emoji first, then check Moodio's output.
    - Discuss any surprises or misclassifications.
4. **Code Walkthrough (15 min)**
    - Show simplified `app.py` logic:
        - Bad word filter → sentiment analysis → emoji output.
    - Explain how TextBlob calculates polarity.
5. **Wrap-Up (5 min)**
    - Recap key points about **text mood detection**.
    - Discuss potential applications in chatbots, games, or learning tools.

Learning Outcomes

- Students can explain **how a program detects mood in text**.
- Students learn **basic NLP concepts** using TextBlob.
- Students understand **safe coding practices** for kid-friendly applications.
- Students can **interact with and test a web app**, connecting code logic to results.