Lesson Plan: Mood2Emoji (Kid-Safe Text Mood Detector)

Target Age Group: 12–16 years
Duration: 60 minutes
Instructor: Prince Kumar

1. Objective

Students will learn how computers can understand human emotions through text. They'll build a simple web app using Python and Streamlit that detects moods from text and shows matching emojis safely for all ages.

2. Learning Outcomes

- Understand basic Natural Language Processing (NLP) concepts.
- Learn how sentiment analysis works in simple terms.
- Write Python code using libraries like TextBlob or Transformers.
- Build and deploy a Streamlit web app.
- Learn responsible and safe AI design for younger audiences.

3. Topics Covered

- What is sentiment analysis?
- How computers detect moods in sentences.
- Introduction to Python libraries (TextBlob, Streamlit).
- Building a simple user interface.
- Safe handling of inappropriate or gibberish input.
- Hosting apps online with Streamlit Cloud.

4. Session Flow (60 Minutes)

- 0–10 min: Introduction to emotion detection and examples.
- 10–20 min: Explain TextBlob and basic sentiment scoring.
- 20–35 min: Write and test simple Python mood detection code.
- 35–50 min: Add emojis and explanations for detected moods.
- 50–60 min: Deploy app on Streamlit and class discussion.

5. Activity Explanation

Each student builds a 'Mood2Emoji' app that takes a short sentence, analyzes its mood, and displays an emoji (e.g., happy, sad, angry, neutral). If the text is meaningless or inappropriate, the app replies kindly that it couldn't understand. Students test their app with creative inputs and share results with the class.

6. Teaching Notes

Encourage creativity — let students test phrases from everyday life. Discuss how AI tools interpret language and why responsible design matters, especially for kids. Explain that models sometimes make mistakes and how developers can improve fairness and accuracy.

7. Tools & Requirements

- Python 3.9+
- Streamlit
- TextBlob / Hugging Face Transformers
- Basic Internet access for deployment
- GitHub account for hosting the project

8. Assessment Criteria

- Functionality of the app (25 pts)
- Code clarity and simplicity (15 pts)
- Logic and safe responses (25 pts)
- Lesson understanding and creativity (20 pts)
- Documentation and reflection (15 pts)