# ARTIFICIAL INTELLIGENCE & MACHINE LEARNING – (Project abstract)

Name: Venigalla Sahithi – 2320030101

Section: S-5

Problem Statement: Development of a Sign Language Recognition System using Python.

Effective communication is vital for the hearing-impaired community, yet many face barriers due to the lack of understanding of sign language. This project aims to create a Sign Language Recognition System that translates sign gestures into text or speech, facilitating better communication.

Objective: Accurately recognizes hand gestures corresponding to a specific sign language (e.g., American Sign Language). Provides a real-time feedback and translation of recognized gestures.

Dataset Creation: Collect and preprocess a diverse dataset of sign language gestures.

## Sign Language Datasets:

Images of hand gestures for the American Sign Language (ASL) alphabet (24 classes).

WLASL (Word-Level ASL):

Contains 2,000 common words in ASL for word-level recognition.

## Algorithm:

1. **Data Collection**

Datasets: Use Sign Language MNIST and WLASL.

Prepare Images**:** Make sure they're the same size and quality.

Labeling: Ensure proper labels for each sample.

2. **Sequence Analysis**

Extract Features**:** Find important parts of the images.

Modeling: Use a special model to understand the sequence of images.

3. **Model Refinement**

Train the model using the data and adjust the model for it to work more efficiently.

4. **Validation**

Measure accuracy, precision and see how often the model gets the right answer.

5. **Visualization**

Confusion Matrix: Visualize predictions vs. actual labels.

## Expected Outcome:

Recognition algorithm include achieving an accuracy of 90% or higher on the test dataset, with high precision and recall values (ideally ≥ 0.85). The model should demonstrate robustness by effectively generalizing to unseen data and recognizing variations in signing styles. It should enable real-time recognition with providing users with immediate feedback. Visual insights like confusion matrices and training curves will highlight performance and learning trends. Lastly,

the system should be easily integrable into applications and scalable for multiple users without performance issues.