# Smart Face Detection System

**ELC Activity Title:**

Real-Time Object Detection and Drawing Effects

**Project Topic:**

Face Detection

**Team Details**

Name:Kriti Sharma  
Roll Number: 102316019  
Contribution: Complete development, testing, and demo video recording.

**Tools & Technologies Used**

- Python 3.x  
- OpenCV (Computer Vision Library)  
- Haarcascade Pre-trained Models  
- Webcam  
- CSV Logging  
- OBS Studio / Clipchamp for Video Recording

**Objective**

The aim of this project is to build a real-time face detection system that not only detects faces but also captures images automatically when a smile is detected. It also provides features like face count display, manual image saving, and smile logging in a CSV file — using Python and OpenCV.

**Key Features Implemented**

1. Real-time face detection using Haarcascade.  
2. Smile detection with auto-image capture.  
3. Live face count overlay.  
4. Manual image capture using keyboard 's' key.  
5. CSV log file to store smile detection events.  
6. Images saved with timestamp-based filenames.

**How the System Works**

- The script starts the webcam and begins reading video frames.  
- Faces are detected and highlighted with blue rectangles.  
- Smile detection is applied to each detected face.  
- If a smile is detected, the frame is auto-saved, and the event is logged.  
- The user can press 's' to manually capture an image.  
- The number of detected faces is dynamically displayed on the video feed.  
- The system stops when the user presses the 'q' key.

**Challenges Faced**

- Preventing repeated auto-captures for continuous smiles.  
- Handling varying lighting conditions for accurate detection.  
- Real-time performance tuning to ensure smooth video feed.

**Conclusion**

This project successfully demonstrates how real-time face and smile detection can be implemented using simple computer vision tools. It offers practical applications in contactless photo capturing, emotion recognition, and real-time monitoring. The project aligns with the goals of the ELC activity by providing hands-on experience with Python and computer vision.