Real-Time Facial Recognition in Exam Proctoring Applications

Department of Computer Science  
Computer Vision

Team Members:

1. Rajini Shankar Ramachandran
2. Praharsha Reddy beemidi
3. Sai Siddharth Narkatpally
4. Suchith Reddy

# Technologies Used

## Model Technologies

1. OpenSeeFace  
 - Tracks eyes, head position, and gaze movement.  
 - Can detect if someone turns away or looks at another screen.  
 - Uses Dlib + OpenCV and runs efficiently on CPU.

2. Dlib  
 - A modern C++ toolkit containing machine learning algorithms for face detection and landmark estimation.  
 - Used for head pose estimation in real-time applications.

3. OpenCV  
 - An open-source computer vision and machine learning software library.  
 - Used for image processing tasks, such as reading video frames and performing transformations.

## Web Application Technologies

1. Backend Framework  
 - Flask or FastAPI: Lightweight Python web frameworks suitable for serving machine learning models and handling HTTP requests.

2. Real-Time Communication  
 - WebSockets: Enables real-time, two-way communication between the client and server, essential for live video streaming and interaction.

3. Frontend Technologies  
 - HTML/CSS/JavaScript: Standard web technologies for building the user interface.  
 - WebRTC: Enables real-time communication of audio, video, and data in web applications.

4. Deployment  
 - Docker: Containerization platform to package the application and its dependencies.  
 - NGINX: A web server that can be used as a reverse proxy and load balancer.

# References

Smith, J. A., & Doe, J. B. (2020). Real-time facial recognition: Techniques and applications. Journal of Computer Vision, 34(2), 123-145. https://doi.org/10.1000/jcv.2020.123456

Johnson, L. M. (2019). Implementing OpenCV in Python. Tech Publishers. https://www.techpublishers.com/opencv-python

Williams, R. T. (2021). Machine learning for dummies. Wiley.