Progress Report:

**Team QuadCore**

**8 July**

**CP-1 1:00 PM**

* Basic setup of react with boilerplate code
* Login page frontend code with mock api calls(backend yet to be done)
* Not Found Page(in case of 404) frontend code

**CP-2 2:00 PM**

* Trained YOLOV8 on Roboflow dataset to detect 2 classes of accidents ‘NormalAccident’ , ‘SevereAccident’.
* Uploaded accident videos for testing

**CP-3 4:00 PM**

* Coded accident.py file , which makes use of the trained accident detection model and the base YOLOV8 to detect the type of the accident and the type of vehicle involved.
* Returns the frame at which the accident took place and the track\_id of the vehicle and saves it to json file.

**CP-4 6:00 PM**

* Completed the frontend for signup page(with mock backend api calls), homepage design
* Also completed police dashboard frontend with mock data

**CP-5 8:00 PM**

* Trained a YOLOv8-based model for automatic license plate detection and number extraction and integrated Optical Character Recognition (OCR) to extract license numbers from detected plates.
* Validated model accuracy with multiple test cases (different vehicle types, lighting conditions).
* In Progress: Integrating this data into the backend (Django dashboard) for police access.

**9th July**

**CP-1 2:00 PM**

-Added hospital dashboard frontend with mock backend api

- Added backend logic for users in django

- ⁠Added code for speed detection in vehicles

-Created flask and react frontend to upload cctv footage for the prototype

**CP-2 4:00 PM**

-Updated hospital dashboard frontend to display image

-Records accident accepted in the hospital dashboards.

**CP-3 11:00 PM**

-Integrated Django backend and Flash backend

-Added map view for easier location

-Completed police dashboard