LANEBOT

Finding the Lane to Smarter Driving.

AKSITHA VIJAYRAJ, ISHA RAO, SREYASRI SURAPANENI, RASHEKA SENTHIL KUMAR

WE HACK

01. PURPOSE

Advanced driving assistance features are often exclusive to newer vehicles, leaving older cars without essential safety support.

LaneBot brings AI-powered lane & object detection to any car through our app.

WE HACK

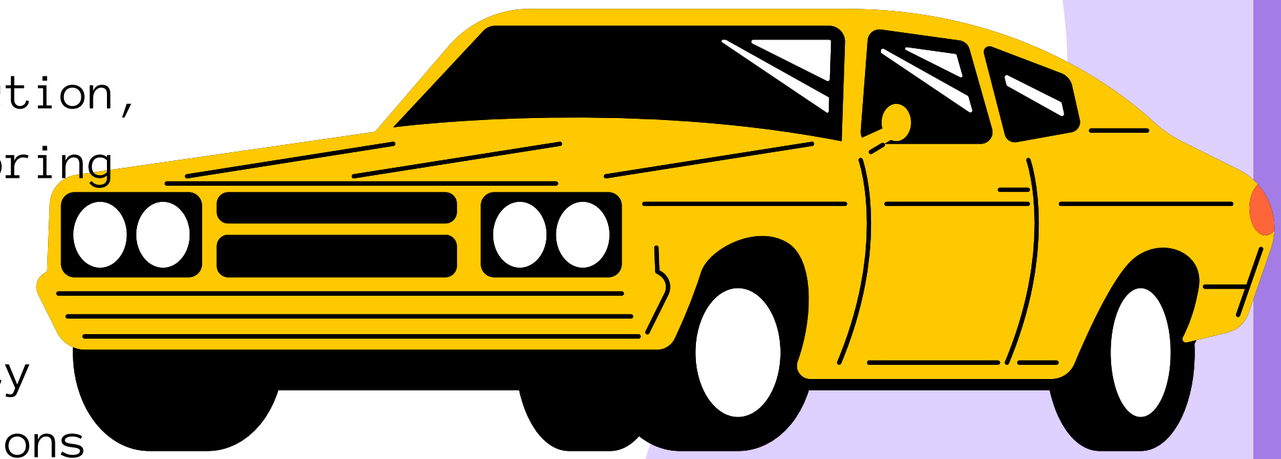
02 INSPIRATION

LaneBot embraces the theme by creating a seamless bridge between past, present, and future driving experiences

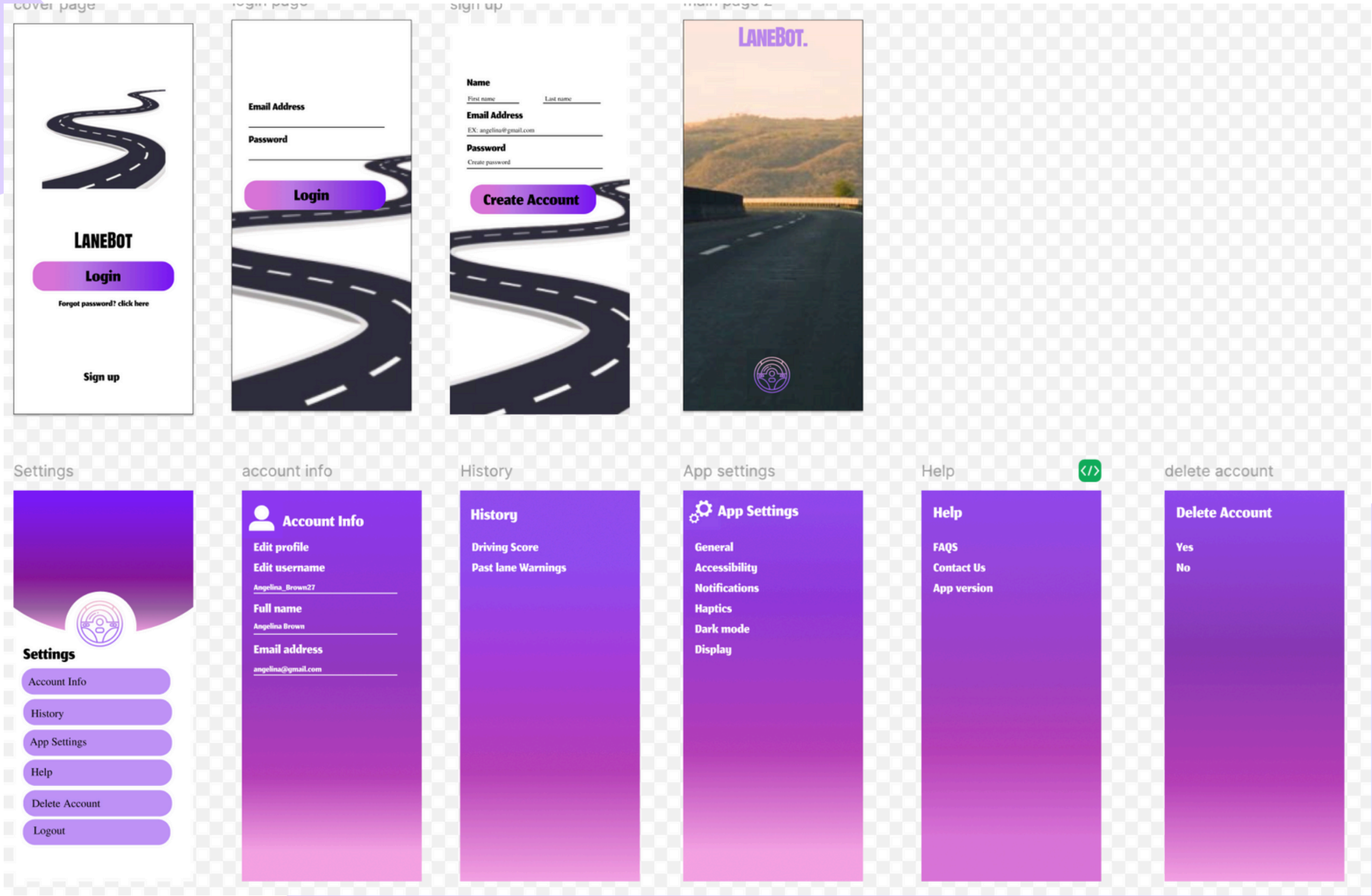
Past: Older cars lack modern safety features. LaneBot brings advanced safety to them, preserving their usability.

Present: Provides real-time lane detection, obstacle alerts, and lane score monitoring for safer driving

Future: Potential for predictive safety features and personalized recommendations to improve driving habits.



OUR UI - FIGMA

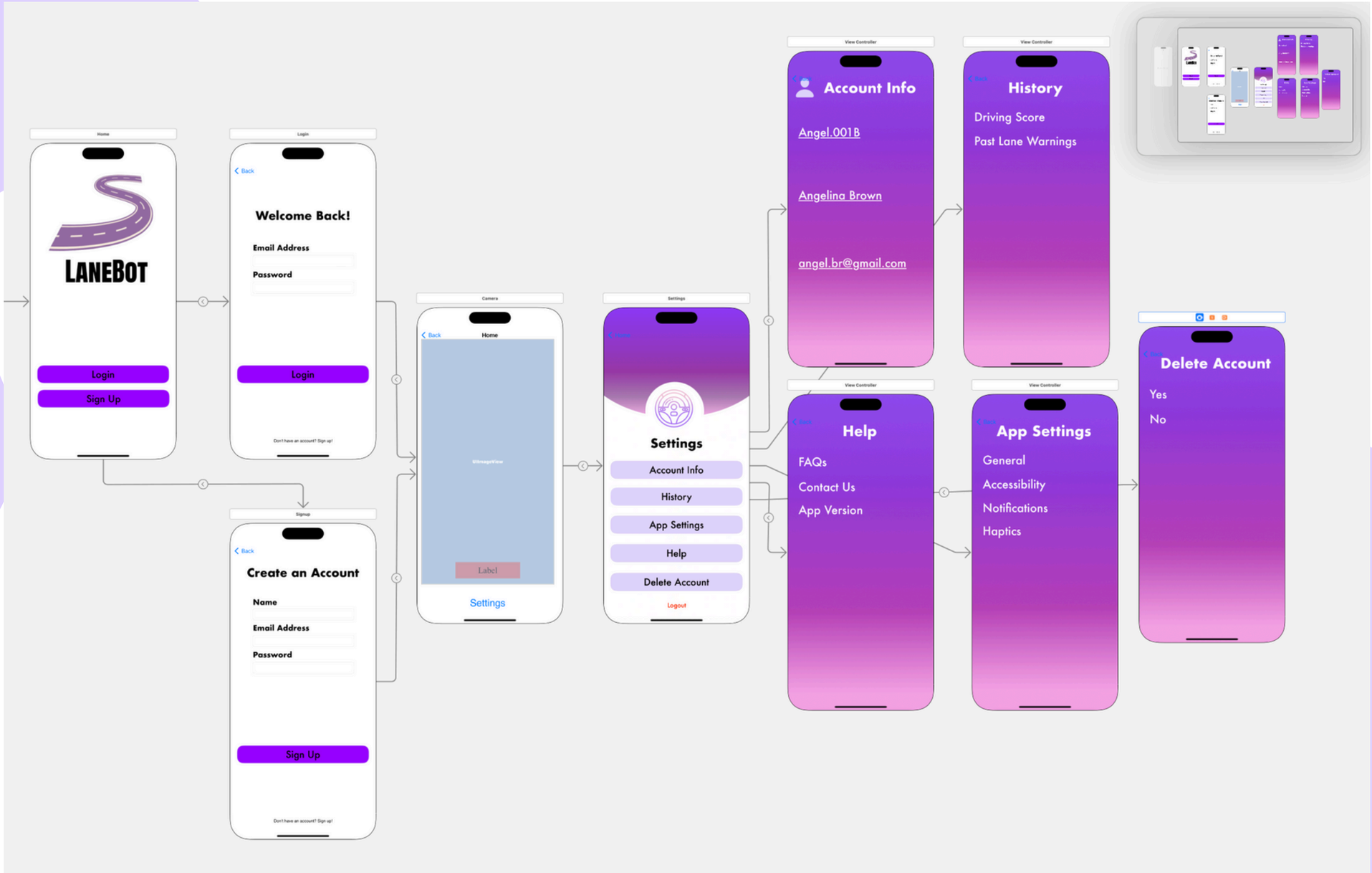


WE HACK

OUR UI - FIGMA

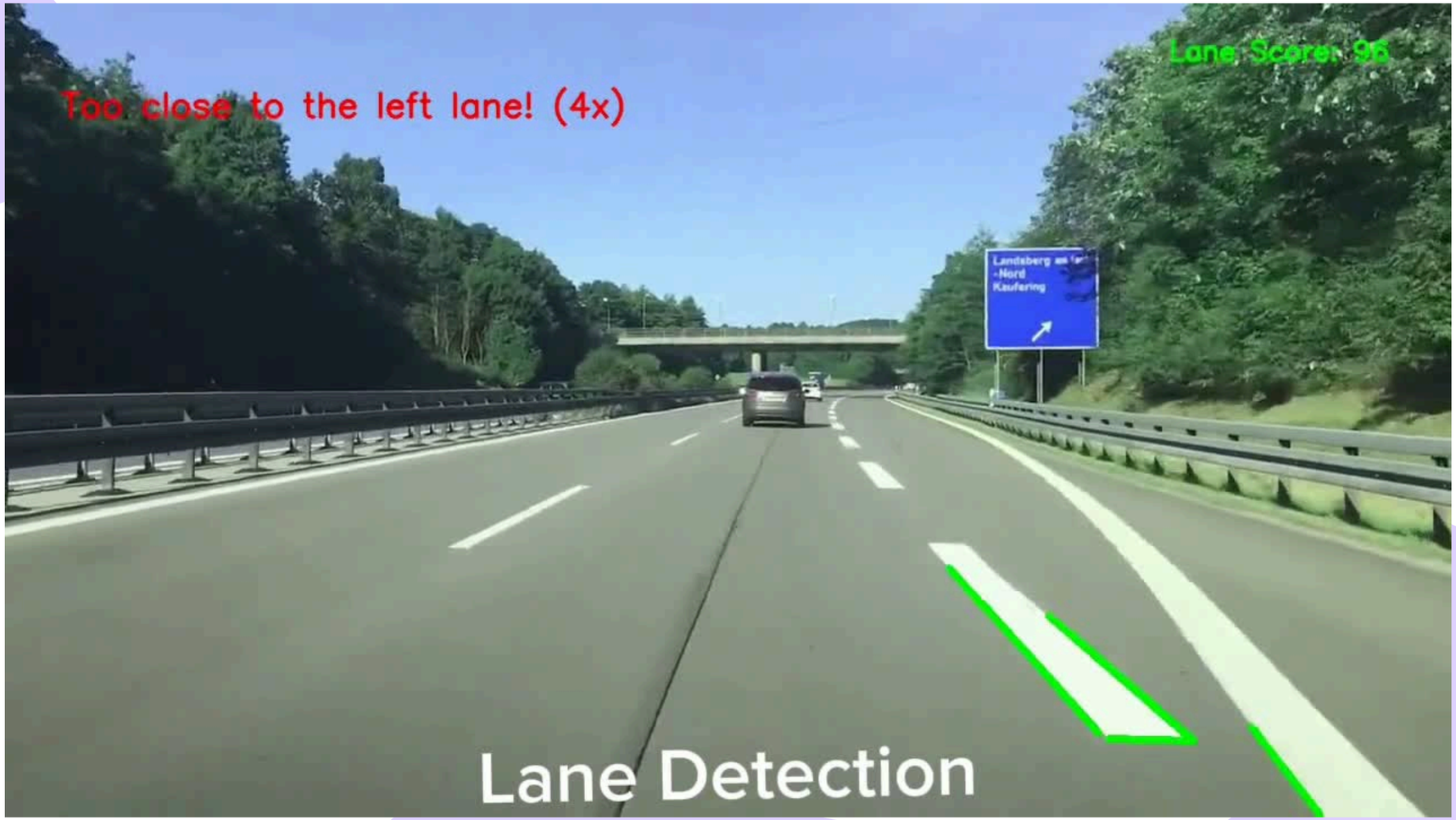
WE HACK

OUR UI - IN SWIFT



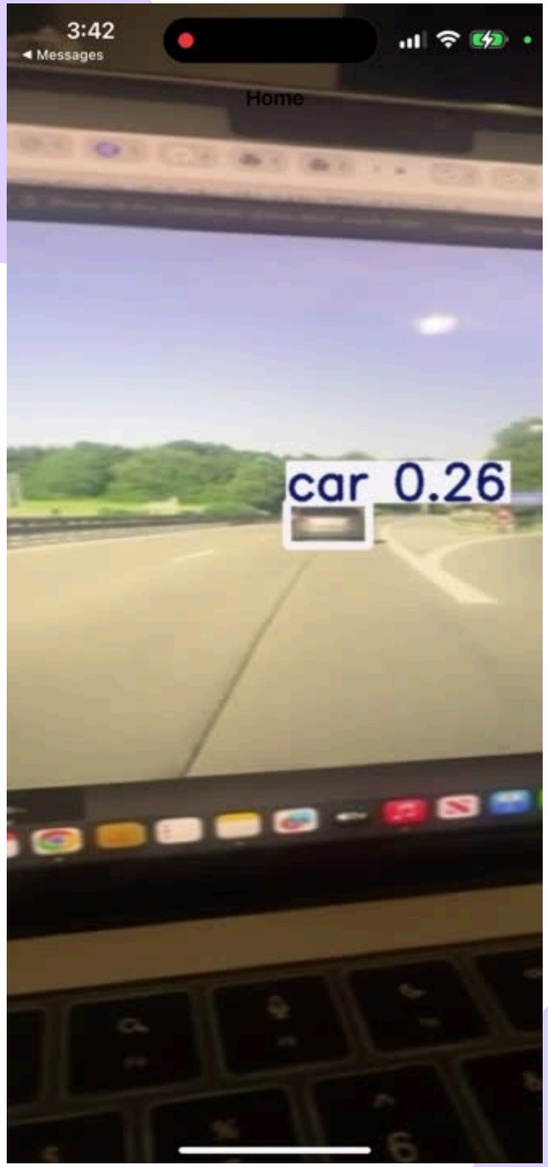
WE HACK

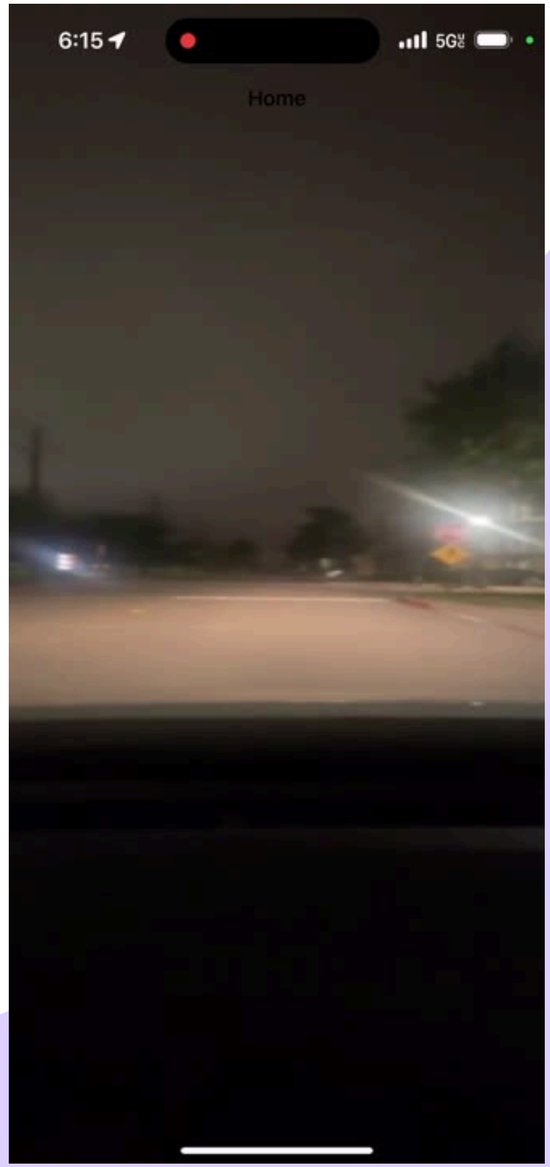
LANE & OBJECT DETECTION



WE HACK

LANE & OBJECT DETECTION





WE HACK

TECH STACK

BACK-END

- Developed using Python
- lane detection using OpenCv
- object detection using YOLOv8
- real time detection via camera in the app
- integration of AI models
- Conducted real world testing to ensure accurate performance under various driving conditions

FRONT-END

- We designed a clean and intuitive user interface on figma
- Designed real time detection to ehance driver awareness
- Focused on user-friendly experience to make the experience simply, faster and efficient using Swift on Xcode

WE HACK

04. LIBRARIES UTILIZED

By leveraging OpenCV for image processing and YOLOv8 for object detection, we enabled real-time lane and obstacle recognition directly from a mobile camera feed

WE HACK

CHALLENGES

- **Swift Integration:** Implementing real time AI detection systems with Swift ensuring compatability with Python models were challenging.
- **UI design**: Creating a functional, visually appealing interface in Figma that effectively displayed real-time data required multiple iterations.
- **Testing & Simulation:** Testing LaneBot in real-world conditions and refining algorithms for accuracy involved rigorous debugging.
- **Feature Prioritization**: Balancing ambitious features with practical implementation within a limited timeframe was crucial.

WE HACK

04. THE FUTURE OF LANEBOT

Fixing Challenges

- Improving lane detection accuracy for all road conditions
- Implementing predictive analysis to anticipate lane drift and risky maneuvers,
- Creating personalized recommendations for safer driving.

Enhanced Functionality

- Improving lane detection accuracy for all road conditions
- Implementing predictive analysis to anticipate lane drift and risky maneuvers,
- Creating personalized recommendations for safer driving.

WE HACK

04. THE FUTURE OF LANEBOT

Features we want to Implement:

- **Driving History Analysis**: Review past trips, track improvements, and get insights on dangerous moments.
- **Predictive Safety Alerts**: Warnings based on common patterns or risky areas identified from previous drives.
- **Future Routes**: Suggesting safer or more scenic paths using previous trip data and real-time traffic analysis.
- **Enhanced UI**: More polished dashboard with real-time visualizations and improved feedback system.

THANK YOU!

WE HACK