Software Engineering in Automotive and Mobility Ecosystems

Project status report no.1

Team : SEA:ME

Prepared by : Minhyeok CHOI

Date : 21/12/2025

Reporting period : 03/11/2025 ~ 20/12/2025

Contents

---

1) Executive Summary

This sprint focused on bringing up the vehicle with the BFMC start-up code, updating key hardware components (main board and camera), verifying communication between ECUs, and enabling ROS-based steering and throttle control.

On the physical vehicle, we achieved reliable boot-up and consistently ran the start-up stack, then verified correct steering and throttle response via ROS topics. We also integrated lane following on the real car and enabled autonomous mode through the dashboard “Auto” button.

In parallel, we implemented and validated a vision-based localization system in the simulation environment.

Next sprint, we will port the localization pipeline to the real vehicle and develop a global path-following system.

---

2) Sprint Goal & Key Deliverables

- [x] Built and executed the BFMC start-up code
- [x] Completed hardware redesign/modification and assembly
- [x] Replaced key components (main board & camera) and verified integration
- [x] Completed steering calibration
- [x] Confirmed ROS 2 communication within the Brain project
- [x] Enabled ROS-based steering and throttle control
- [x] Implemented lane detection and lane following on the real vehicle
- [x] Integrated autonomous mode via the dashboard “Auto” button
- [x] Set up the simulation environment and development workflow
- [x] Implemented and validated a vision-based localization system in simulation

3) What we built & Current phase

- Hardware (Modification & Mounting) (Repo : seame_hardware)

We replaced the vehicle's original main board and camera with a Jetson Orin Nano and a RealSense D455f. To mount the upgraded components securely, we designed custom parts in Siemens NX and fabricated them using 3D printing. This setup minimizes camera vibration during driving and ensures stable sensor mounting.

- Software (Integration & Feature Implementation)

<u>Vehicle bring-up and run workflow.</u> (Repo : seame_brain)

We validated the BFMC start-up stack on the physical vehicle and streamlined the code/services and run procedure so the same bring-up process can be reliably reproduced on the Jetson Orin Nano. We also adapted the Brain project for the Jetson environment, improved the dashboard UI, and refined the image-processing pipeline to enable stable real-time monitoring with the replaced camera.

<u>ROS-based control interface.</u> (Repo : seame_brain)

We extended the system to exchange data with the Brain project via ROS topics and built a new process that enables steering and throttle control.

<u>Real-vehicle lane-following integration.</u> (Repo : seame_ros)

We implemented lane detection and lane following on the real car and integrated it with the Brain project so autonomous mode can be triggered via the dashboard "Auto" button. To handle camera frame drops and interruptions, we applied a Kalman filter–based lane estimation approach. Based on this, we built an end-to-end driving pipeline: local path generation → planning node → control node**. In addition, we implemented YOLO-based object detection** using the real vehicle's camera input.

<u>Simulation development pipeline.</u> (Repo : seame_simulation)

We set up the simulation environment and implemented/validated a camera vision–based localization system. We improved estimation accuracy using a Kalman filter, establishing a foundation that can later be extended to global navigation.

- Current Phase

The physical vehicle bring-up is complete: the car can be controlled via ROS, and lane following can be executed through the dashboard auto mode. We are currently integrating the object-detection node into the autonomous driving stack while optimizing the system to reduce CPU usage. In simulation, vision-based localization has been validated, and in the next sprint we will port it to the real vehicle to verify accuracy and robustness.

---

4) Next step

Now that our development environment is fully set up, the next sprint will focus on designing and implementing an efficient global path generation strategy to cover the track's waypoints. We will also port the vision-based localization validated in simulation to the real vehicle and achieve stable global path following in the next step. Finally, we will integrate server communication to transmit the vehicle's estimated position accurately and consistently for monitoring and coordination.