

Audrey Luan

Stephen C. Petersen

ECE 129A - Capstone pt 1

27 October 2025

C.A.R.T. - Carry Assist Robotic Transport

Background:

This project is supposed to help everyone in the University of California move heavy loads across campus and soft ground sections, due to lack of available parking for vehicles and the mountainous terrain. This will make our university more accessible for the elderly and disabled individuals, and make life easier for the students and faculty alike. The deliverable might be used for research (HARE lab involvement will affect the project outcome), but it will definitely be used by the UCSC community.

Relevant references and links -

- <https://clearpathrobotics.com/robots/>
- <https://www.starship.xyz/>

Problem Statement:

There is a large gap that large vehicles such as cars and semi trucks can't provide on the UCSC campus for moving heavy loads. The main challenge is to have a robotic cart that is durable enough to carry a 10-50 pound load up a steep incline. The goal is to transport loads of goods to reach different destinations across campus without putting significant strain on their bodies.

Image: [starship project from UC Irvine]



Target End User/Client Profile:

The end users of this product are UCSC students (campus-wide survey needs to be done), research labs, faculty, visitors, and university clubs.

Team Composition:

- Team structure will be determined based on final team size and project requirements.
- The project scope is predominantly mechanical engineering and hardware-focused.

High-Level Potential Solutions:

- (to be added during brainstorming)

Project Timeline and Major Milestones:

- 6 month timeline

Month 1: Brainstorming

Month 2: Component sourcing and ordering the hardware parts

Month 3: Concurrent subsystem development

- Mechanical: Chassis prototyping and fabrication
- Software: Finite state machine architecture and implementation
- Electronics: Embedded system integration and sensor interfacing

Month 4: Validation and verification phase including software debugging, hardware stress testing, and mechanical iteration.

Month 5-6: System integration, performance benchmarking, and extension objectives (if we've already completed the baseline goals)