AnyWill

Autonomous Wheelchair

Team A: Final Report

Haoyang He
Yu-Hsin Chan
Chiawen Liao
Chao-I Tuan
Sonic Kuo

Sponsors: Cherie Ho, Sebastian Scherer

Dec 10, 2025

Abstract

Uneven, cluttered sidewalks and constant steering impose significant physical and cognitive load on wheelchair users. This report presents *AnyWill*, an outdoor, user-interactive autonomous wheelchair designed to provide end-to-end mobility on urban sidewalks while respecting individual preferences.

The system retrofits a commercial powered wheelchair with a ZED-X RGB-D camera, dual-antenna RTK GNSS/IMU, and a Jetson AGX Orin running ROS 2. Visual and geometric features from first-person images and point clouds are fused into a learned bird's-eye-view costmap using an inverse reinforcement learning model trained on expert teleoperation, capturing subtle terrain semantics such as curbs, rough pavement, and compliant surfaces. On top of this costmap, a global planner uses an OpenStreetMap-based wheelchair-accessible map, and a model predictive path integral local planner replans at 2 Hz to generate safe, comfortable trajectories. A companion iOS app allows users to set destinations, monitor progress, and optionally invoke a vision-language model planner that proposes semantically rich routes from first-person imagery for user approval. Safety is enforced through multiple hardware E-STOPs, a dedicated safety circuit, and conservative speed limits.

In a 450 m Fall Validation Demonstration on real sidewalks, AnyWill successfully completed the route with a 1.2 m final error, avoided all static and dynamic pedestrians, and required only a single brief teleoperation during GNSS degradation, demonstrating the technical feasibility of user-interactive autonomous sidewalk mobility.

Contents

1 Project Description **1**

2 Use Case **1**

3 System level requirements **3**

4 Functional Architecture **5**
4.1 Power Management 5
4.2 User Command Interpretation 5
4.3 Environmental Perception 6
4.4 Localization and Mapping 6
4.5 Navigation and Planning 6
4.6 Motion Execution 6
4.7 Safety and Emergency Handling 6
4.8 Mission Completion 6

5 Trade Studies **7**
5.1 System Level Trade Study 7
5.2 Subsystem-Level Trade Study 7
5.2.1 Electric Wheelchair Purchase Options 7
5.2.2 Perception Method Design Options 8
5.2.3 RGB-D Camera Options 8
5.2.4 Localization Sensor Suite Trade 9

6 Cyberphysical Architecture **10**
6.1 User Interface 10
6.2 Sensing 10
6.3 Perception, Localization & Mapping 11
6.4 Navigation & Planning 11
6.5 Powertrain Management 11
6.6 Mechanical 11

7 System Description and Evaluation **12**
7.1 Overall System Depiction 12
7.2 Subsystem Descriptions 12
7.2.1 Mechanical 12
7.2.2 Powertrain 12
7.2.3 Sensing 14
7.2.4 Perception 14
7.2.5 Planning 15
7.2.6 User Interface 17
7.3 Modeling, Analysis, Testing 18
7.3.1 Power Output for AGX Orin 18
7.3.2 Camera Placement 18
7.3.3 Autonomy Testing in IsaacSim 19
7.3.4 Autonomy Testing with ROSBag Playback 20
7.4 FVD Performance Evaluation 20
7.4.1 Demonstration Objectives 20
7.4.2 Success Criteria 21
7.4.3 Performance Results 22
7.5 Strong / Weak points 22
7.5.1 Strengths 22
7.5.2 Weaknesses 22

8 Project Management **23**
8.1 Schedule 23
8.1.1 Planned and Adjusted Schedule 23
8.1.2 Evaluation of Scheduling Process 24
8.2 Budget 25
8.2.1 Budget Analysis 25
8.2.2 Evaluating the Budgeting Process 25
8.3 Risk Management 26
8.3.1 Risk Summary and Mitigation: 27

9 Conclusions **28**
9.1 Lessons Learned 28
9.1.1 Early System Integration Is Critical. 28
9.1.2 Well-Maintained Documentation Aligns the Team and Accelerates Collaboration. 28
9.1.3 Multiple Dry Runs Under Diverse Conditions Reveal System 28
9.2 Future Work 28
9.2.1 Robustifying the System 28
9.2.2 Diverse User Testing 29
9.2.3 Market Research 29

1 Project Description

Autonomous wheelchairs have the potential to greatly enhance mobility and independence, but reliable navigation in real urban environments is still very challenging [1, 2]. Outdoor sidewalks contain many subtle and uncomfortable hazards such as cracked pavement, broken curbs, uneven slabs, puddles, leaves, and temporary obstacles. Simple obstacle avoidance is not sufficient. For example, a wheelchair must distinguish between objects with similar height but very different meaning, such as traversable tall grass and rigid cinder blocks that must not be crossed.

Today, human operators still need to select the most traversable paths, because they can interpret small visual cues that current systems often miss. Our project aims to give wheelchairs similar judgment, enabling them to autonomously assess the environment and choose paths that are both safe and comfortable.

In addition, autonomous wheelchairs operate in a much more varied setting than autonomous cars. Cars mainly drive on structured roads and follow clearly defined traffic rules for most behavioral decisions. In contrast, an autonomous wheelchair must move through sidewalks, curb cuts, park trails, building entrances, indoor and outdoor transitions, and interact with many different types of terrain and objects.

The goal of this project is to build a wheelchair that can autonomously navigate to a user specified destination, safely handle real world hazards, and adjust its behavior according to user preferences provided through audio input. Ultimately, we aim to advance autonomous wheelchair navigation and make urban environments more accessible and comfortable for wheelchair users.

2 Use Case

To illustrate the intended application of our system, we consider the following use case in which Mrs. D uses our autonomous wheelchair to travel from her house to her friend's home (as shown in Fig. 1).

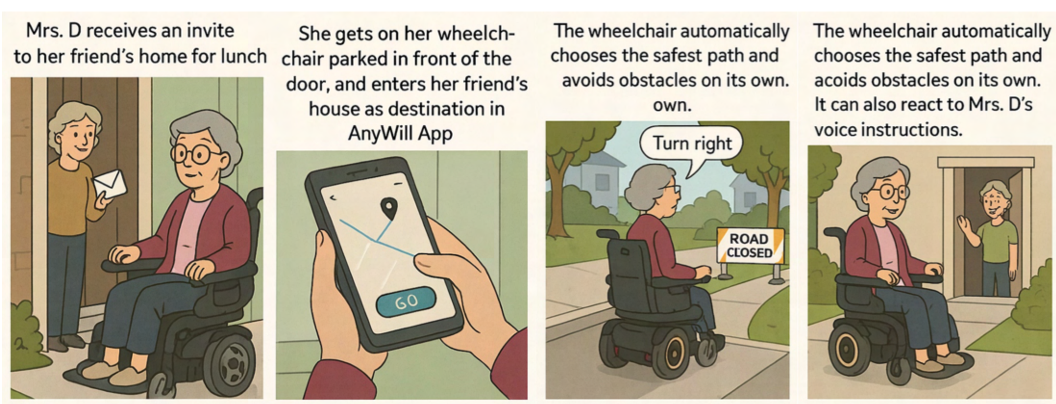


Figure 1: An example use case of Mrs.D going to her friend's place.

Mrs. D receives an invitation to her friend's home for lunch. She gets onto her wheelchair, which is parked in front of her door, opens the AnyWill app on her iPhone, and selects her friend's house as the destination. The wheelchair confirms the goal and then begins to autonomously navigate toward it.

As it moves through the neighborhood, the wheelchair continuously perceives the environment, builds a real-time costmap of the surrounding terrain, selects safe and comfortable paths, and avoids obstacles on its own. The system reasons not only about collision avoidance, but also about user comfort and terrain semantics.

For example, near her friend's house, the system recognizes both a smoother lawn and a bumpier pebble path. Because Mrs. D has previously indicated a preference for staying on paved surfaces, the wheelchair chooses the pebble path to the front door instead of cutting across the grass. Along the way, the perception system detects and avoids static obstacles such as signs, trees, and utility poles, and it can distinguish curb cuts from curb edges so that the wheelchair safely enters and exits crosswalks. It also tracks moving pedestrians and adjusts its path and speed to maintain a comfortable clearance. In addition, the wheelchair can react to Mrs. D's voice commands. When it encounters a "Road Closed" sign, Mrs. D can simply say "Turn right," and the wheelchair reroutes accordingly while still respecting the overall navigation goal.

Throughout the trip, the wheelchair remains fully autonomous in its navigation decisions while still allowing Mrs. D to influence the route through simple voice commands and personalized comfort preferences. Mrs. D arrives safely and comfortably at her friend's front door and happily recommends how convenient and reliable the autonomous wheelchair is. This use case demonstrates how our system integrates perception, planning, hazard handling, and user preference modeling to provide end-to-end autonomous mobility in a realistic urban environment.

While this example focuses on Mrs. D, the intended users of our system are not limited to a single persona. Autonomous wheelchairs can benefit a wide range of people, including older adults with reduced mobility, individuals recovering from leg or foot injuries, and users who experience fatigue or limited stamina during longer trips. By reducing both the physical effort of propulsion and the cognitive load of continuous steering and obstacle avoidance, our system aims to make everyday outdoor mobility more accessible and less taxing for diverse user groups.

3 System level requirements

To evaluate the wheelchair's capabilities, we define specific system-level performance targets that are directly tied to their corresponding functional requirements. These targets are then systematically allocated to individual subsystems, ensuring that each subsystem has clear responsibilities that collectively satisfy the overall system objectives. For each subsystem, we also derive key quantitative metrics that serve as measurable benchmarks and support rigorous performance verification and validation.

All system-level requirements are summarized in the tables below. Table 1 presents the mandatory and desired functional and performance requirements, together with brief descriptions. Table 2 outlines the non-functional requirements that constrain aspects such as usability, safety, and reliability. Table 3 shows how these requirements are allocated to subsystems, highlighting the traceability from high-level system goals down to specific subsystem responsibilities.

Table 1: Mandatory Functional and Performance Requirements

Functional	Performance	Description
F.M.1 Localize itself	**M.P.1.1** Estimate position with error $\leq$ 5m **M.P.1.2** Update pose at 5Hz	The robot must maintain an accurate and timely estimate of its own position to enable effective navigation and task execution.
F.M.2 Map environment	**M.P.2** Generate reasonable cost map within 100ms	The system must be capable of environmental perception and generate cost maps in real time to support safe navigation.
F.M.3 Plan path	**M.P.3** Plan lowest cost path within 100ms after each cost map update	Fast and reliable path planning is essential for autonomy in dynamic environments.
F.M.4 Actuate the wheel	**M.P.4.1** Wheel motor commands update at 10Hz **M.P.4.2** Achieve smooth motion with desired speed $\leq$ 1m/s	The system must issue timely and stable actuation commands to ensure responsive and smooth movement.
F.M.5 Detect static obstacles	**M.P.5.1** Avoid curbs of height $\geq$ 5cm **M.P.5.2** Avoid static humans and safety cones **M.P.5.3** Number of E-STOP safety interventions $\leq$ 2	The robot should safely navigate around common static obstacles encountered in pedestrian spaces.
F.M.6 Adjust path dynamically	**M.P.6.1** Re-plan path within 200ms after detecting new obstacle **M.P.6.2** Number of collisions with moving obstacles = 0	The robot must dynamically adapt its path in response to changes in the environment to avoid collisions.
F.M.7 Afford payload	**M.P.7** Carry user and payload of at least 200lb	The robot must support users and their belongings to ensure real-world usability and comfort.
F.M.8 Identify safe zones	**M.P.8** Detect and mark safe zones with 90% accuracy	The system must reliably identify safe locations for stopping, pickup, or drop-off operations.
F.M.9 Ensure emergency stop	**M.P.9** Stop within 300ms after E-Stop activation	The robot must respond rapidly and reliably to emergency stop signals to ensure user and bystander safety.

F.M.10 Reach Destination	**M.P.10.1** Reach goal within drift $\leq$ 5m **M.P.10.2** Number of teleoperation interventions $\leq$ 1 **M.P.10.3** Teleoperation distance $\leq$ 5m **M.P.10.4** Complete 0.25 km traversable route $\leq$ 8 min	The system must reliably reach the desired destination within acceptable error margins, minimal teleoperation, and reasonable time constraints.
F.D.11 Climb slopes such as sidewalks	**D.P.11** Traverse sidewalk $\leq$ 15 degrees	Maintain traction and stability when ascending or descending typical sidewalk ramps with slope up to 15°.
F.D.12 Detect and avoid dynamic obstacles	**D.P.2.1** Replan trajectories $\leq$ 50 ms **D.P.2.2** Consider moving pedestrian $\geq$ 2	Continuously sense moving agents (for example pedestrians or cyclists) and update the planned trajectory within 50 ms.
F.D.13 Receive commands from the user	**D.P.3** Success rate $\geq$ 70%	Correctly interpret and execute high–level user commands (such as turn, or change destination) in at least 70% of attempts under typical outdoor conditions.

Table 2: Mandatory Non-Functional Requirements

Requirements	
M.N.1	Compact and lightweight for easy navigation
M.N.2	Reliable performance ensuring consistent, smooth operation with payload
M.N.3	Safety constraints include a maximum speed of 3–5 mph and a range of 10 miles on a single charge
M.N.4	Modular design to facilitate easy maintenance and upgrades
M.N.5	Cost-efficient design, priced at 150–200% of standard non-autonomous wheelchairs
M.N.6	Generate wheelchair-traversable paths via intuitive user interface

Table 3: Subsystem-Level Functional and Performance Requirements

Subsystem	**Functional Requirement**	**Performance Requirement**
Mechanical	**F.M.7** Afford payload	**M.P.7.1** Carry payload $\leq$ 200 lb
Robotic Software System Navigation	**F.M.6** Preplan safe path to go	**M.P.6.2** Reach every navigation waypoint $\leq$ 5 m
Robotic Software System Local Planning	**F.M.5** Detect static obstacles **F.M.6** Adjust path dynamically	**M.P.6.3** Replan trajectories $\leq$ 50 ms
Robotic Software System Perception, Localization & Mapping, Planning, Navigation, Controls	**F.M.5** Detect static obstacles **F.M.6** Adjust path dynamically	**M.P.6.4** Avoid curbs of height $\geq$ 5 cm **M.P.6.5** Avoid potholes of radius $\geq$ 5 cm and depth $\geq$ 10 cm
Powertrain – E-Stop	**F.M.9** Ensure emergency stop	**M.P.9.2** React to emergency stop $\leq$ 50 ms

4 Functional Architecture

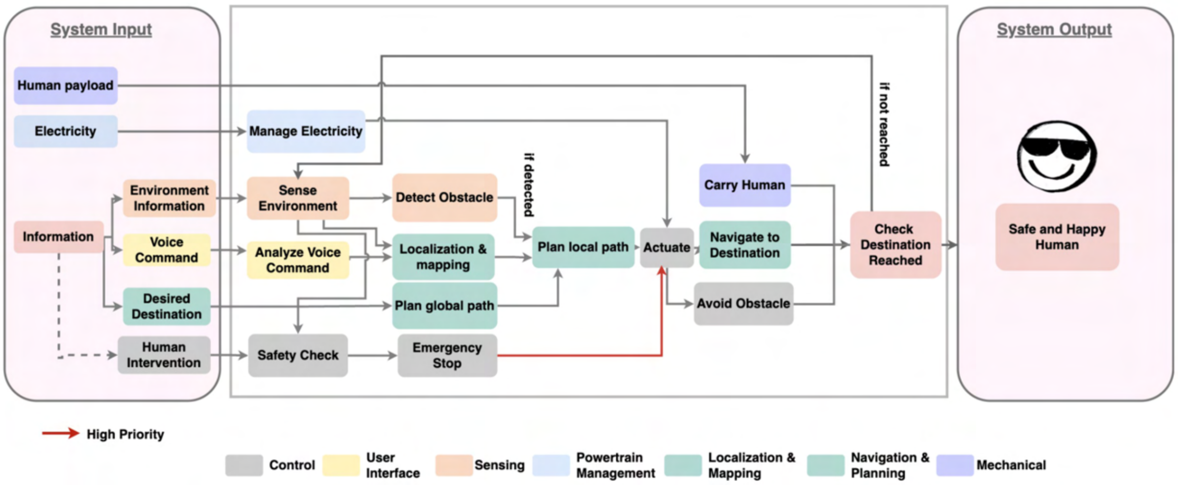


Figure 2: Functional architecture.

The functional architecture in Figure 2 illustrates the high level functions of the system and the flow of information from inputs to outputs. The wheelchair receives two main classes of inputs: **Physical Input** and **Information Input**.

Physical Input consists of the human payload and electrical power, which are required for both mission execution and basic system operation. **Information Input** includes the user's desired destination, voice commands, environment information from onboard sensors, and optional human intervention through the joystick or other manual controls.

These inputs propagate through a set of interconnected functional blocks that allow the system to perceive the environment, interpret user intent, generate plans at both global and local levels, and command the actuators that move the wheelchair. The architecture highlights closed loop behavior, explicit safety supervision, and an emergency path that can interrupt normal operation when needed. The system output is a safe and comfortable ride that transports the user to the goal while keeping them informed about the current state.

4.1 Power Management

The *Manage Electricity* block distributes power from the batteries to all subsystems and monitors overall power availability. It ensures that sensing, computation, communication, and actuation have sufficient and stable power to operate throughout the mission.

4.2 User Command Interpretation

The *Analyze Voice Command* module converts spoken language into structured navigation or mode commands. Together with the *Desired Destination* input and *Human Intervention*, it determines the current mission goal and any user initiated overrides.

4.3 Environmental Perception

The *Sense Environment* function gathers data from onboard sensors, such as cameras and depth sensors, and forwards processed information to *Detect Obstacle*. This enables the system to recognize static and dynamic obstacles, curbs and curb cuts, and other salient features in the environment.

4.4 Localization and Mapping

The *Localization & mapping* block fuses odometry, GNSS, and visual information to estimate the pose of the wheelchair and maintain an internal map. This information supports both global route planning and local maneuver decisions.

4.5 Navigation and Planning

Using the destination and the map, the system first computes a *Plan global path* from the current pose to the goal. In parallel, the *Plan local path* function refines this route in real time based on updated sensor data and detected obstacles. Local planning is responsible for obstacle avoidance, following the global route when possible, and replanning when blocked.

4.6 Motion Execution

The *Actuate* block converts planned velocities and trajectories into motor commands. *Carry Human* and *Navigate to Destination* represent the mechanical and control functions that physically move the wheelchair and track the desired path. The *Avoid Obstacle* function modifies commands at a lower level if necessary to maintain a safe clearance from obstacles.

4.7 Safety and Emergency Handling

Safety Check monitors subsystem status, sensor health, and user intervention. If an unsafe condition is detected, or if the user requests an immediate stop, the *Emergency Stop* path bypasses normal navigation and triggers a high priority command that halts actuation and brings the wheelchair to a safe stop.

4.8 Mission Completion

The *Check Destination Reached* block evaluates whether the wheelchair has arrived within the desired vicinity of the goal. Once the destination is confirmed, it terminates the navigation task and reports success to the user, closing the loop with the intended system output of a safe and happy human.

This functional architecture provides a clear mapping between user inputs, core perception and planning functions, safety supervision, and the final outcome, and it serves as the basis for allocating requirements to subsystems and designing corresponding software and hardware modules.

5 Trade Studies

We conducted trade studies to understand the system-level and subsystem-level implementation methods of various functional blocks listed in our functional architecture. All trade studies were graded on a scale from 1 - 5, with 1 being inadequate and 5 being excellent for the task in context.

5.1 System Level Trade Study

The primary evaluation criteria in our system-level trade study are safety and convenience, as we believe a high-quality wheelchair should ensure user safety while being easy to use. To determine the scores for each criterion, we conducted research on prior art, including use cases, prior surveys and customer feedback, to assign appropriate scores to each wheelchair type.

Table 4: System Level Trade Study

				
Value Rating	Spec	Wheelchair	Electric Wheelchair	Autonomous Wheelchair
1: Inadequate	Price(USD)	$150-700	$3,000-6,000	$4,500-8,000
2: Tolerable	Cost of Operation	$20/hr to hire someone	$1/hr	$1/hr
3: Adequate				
4: Good				
5: Excellent				
Criteria	Weight Factor (100%)		Value (1-5)	
Safety	35	3	4	4
Convenience	25	2	3	5
Comprehensive Usage	15	5	4	4
Accessibility	15	2	3	5
Affordability	10	4	5	3
Weighted Average	**5**	**3**	**3.7**	**4.3**

5.2 Subsystem-Level Trade Study

We conducted four subsystem-level trade studies to evaluate design and purchase options:

5.2.1 Electric Wheelchair Purchase Options

We performed a trade study over four candidate powered mobility bases (WHILL C2, Jazzy 600 ES, Robooter E60 Pro, a generic 4-wheel mobility scooter, and a lightweight manual chair) since our approach is to retrofit an existing platform with autonomous capabilities. As summarized in Table 5, each option was scored against weighted criteria derived from our requirements. Performance-related criteria include top speed, payload capacity, and all-terrain capability, while non-functional criteria capture affordability, safety, ease of retrofitting (mechanical mounting space, electrical access, ROS compatibility), rider comfort, and delivery lead time.

Safety and ease of retrofitting received the highest weights because they directly affect both user risk and the feasibility of our integration work. Based on the resulting weighted averages, the Jazzy 600 ES achieved the best overall score, offering a good balance of payload, stability, and modifiability at a moderate cost. We therefore selected it as the base platform for our autonomous wheelchair prototype.

Table 5: Subsystem-Level Trade Study - Electric Wheelchairs

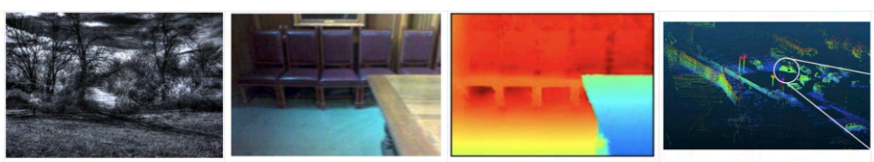
	Spec	WHILL C2	Jazzy® 600 ES	Robooter E60 pro	4-wheel Mobility Scooter	BALAMI Lightweight
	Front Wheel Diameter	9.8"	6"	12"	7.4"	9.8"
	Rear Wheel Diameter	10.5"	14"	12"	7.4"	22.1"
Value Rating	Max Speed	8 km/h	6.43 km/h	10 km/h	6.43 km/h	6 km/h
1: Inadequate	Climbing Angle	10°	15°	9°	8°	8°
2: Tolerable	Suspension	4-wheel	4-wheel	Front wheel	None	None
3: Adequate	Weight	52.8 kg	97.5kg	35.6 kg	40.8kg	17.5 kg
4: Good	Max Loading	136kg	136kg	150kg	136 kg	100 kg
5: Excellent	ROS Compatability	Yes	Yes	No	No	No
	Cost	$9,749	$3,739	$4,299	$599	$7,583
Criteria	Weight Factor (100%)			Value (1-5)		
Performance Requirements						
Speed	5	5	4	5	4	3
Payload	10	5	5	5	5	4
All-terrain	15	4	3	4	2	4
None-Functional Requirements						
Affordability	15	1	4	3	5	1
Safety	20	4	3	4	2	4
Ease of retrofitting	20	5	5	3	1	2
Comfort level	5	5	3	4	3	4
Delivery ETA	10	3	5	4	5	5
Weighted Average	**5**	**3.85**	**4**	**3.8**	**3**	**3.2**

5.2.2 Perception Method Design Options

We evaluated four perception options for our system—monochrome camera, RGB camera, RGB-D sensor, and LiDAR (Table 6). Each was scored on perception range, RGB and depth resolution, frame rate, field of view, computation cost, budget, and outdoor compatibility, with computation cost weighted highest because all perception must run onboard.

Monochrome and RGB cameras are cheap and lightweight in computation but lack depth, which makes obstacle and curb detection difficult. LiDAR offers excellent range and accuracy but is too expensive and compute-intensive for our platform. RGB-D sensing provides the best balance of depth quality, cost, and onboard compute, achieves the highest weighted score, and is therefore selected as our primary perception method.

Table 6: Subsystem-Level Trade Study - Perception Methods



Criteria	Weight Factor (100%)	Monochrome Camera	RGB	RGB-D	LiDAR	
perception range	10	4	3	3	5	
rgb resolution	15	1	5	5	1	Value Rating
depth resolution	15	1	1	4	5	1: Inadequate
frame per second	10	5	4	4	4	2: Tolerable
field of view	10	5	4	4	5	3: Adequate
computation cost	20	4	4	3	2	4: Good
budget	10	5	4	3	2	5: Excellent
outdoor compabiblity	10	1	2	4	5	
Weighted Average	**5**	**3.1**	**3.4**	**3.75**	**3.4**	

5.2.3 RGB-D Camera Options

Lastly, we conducted a trade study of four RGB-D cameras (RealSense D435i, RealSense D415, ZED X Stereo, and Orbbec Astra+) to select the most suitable depth sensor for our system. As summarized in Table 7, each candidate was scored against weighted criteria including cost, RGB and depth resolution/FPS, depth range, IMU availability, interface, and outdoor robustness, reflecting our requirement that the wheelchair perceive its environment reliably in outdoor settings. The ZED X Stereo achieved the highest weighted average score, driven by its strong depth resolution, high frame rate, and superior outdoor performance, so we chose it as the primary perception sensor for our autonomous wheelchair.

Table 7: Subsystem-Level Trade Study - RGB-D Cameras

					
Criteria	Weight Factor (100%)	RealSense D435i	RealSense D415	ZED X Stereo	Orbbec Astra+
cost	10	3	4	1	5
rgb resolution/FPS	20	3	3	5	4
depth range	15	3	2	5	4
depth resolution/FPS	20	4	4	5	3
IMU	10	5	0	5	0
Interface	10	5	4	5	3
outdoor ability	15	4	4	5	3
Weighted Average	**5**	**3.75**	**3.1**	**4.6**	**3.25**

5.2.4 Localization Sensor Suite Trade

We evaluated four localization sensor suites for our autonomous wheelchair: iPhone only, iPhone plus ZED odometry, single-antenna RTK GNSS for global position with the iPhone providing geomagnetic heading, and dual-antenna RTK GNSS for both position and heading (Table 8). iPhone-based options are cheap and easy to integrate, but phone-grade GNSS is only meter-level and magnetometer heading is noisy and drifts, even when fused with ZED odometry, which is insufficient for sidewalk-level navigation and stable costmap alignment. Single-antenna RTK GNSS greatly improves (x, y) accuracy to the centimeter level, but still relies on drifting iPhone heading, leading to inconsistencies between the global frame and the local costmap. Dual-antenna RTK GNSS achieves the highest score, providing centimeter-level position and reliable heading even at low speed or while stationary, resulting in a much more stable global frame and safer sidewalk navigation.

Based on this trade study, we select the RTK GNSS + dual-antenna configuration as our localization sensor suite, using it as the primary source of position and orientation for the autonomous wheelchair.

Table 8: Subsystem-Level Trade Study - Localization Sensor Suites

	Spec	iPhone only	iPhone + ZED Odom	RTK GNSS + iPhone	RTK GNSS + dual antenna	
Criteria	Weight Factor (100%)	Value (1-5)				
Position accuracy	25	2	2	5	5	Value Rating
Heading / low-speed orientation	20	2	3	3	5	1: Inadequate
Outdoor reliability	15	1	3	3	4	2: Tolerable
Integration complexity	15	4	3	3	2	3: Adequate
Cost	15	5	3	3	2	4: Good
Power / form factor	10	3	3	3	3	5: Excellent
Weighted average score	**5**	**2.7**	**2.75**	**3.5**	**3.75**	

6 Cyberphysical Architecture

The cyberphysical architecture adheres to the sequential flow established in the system's functional architecture. Figure 3 provides an overview of all system modules.

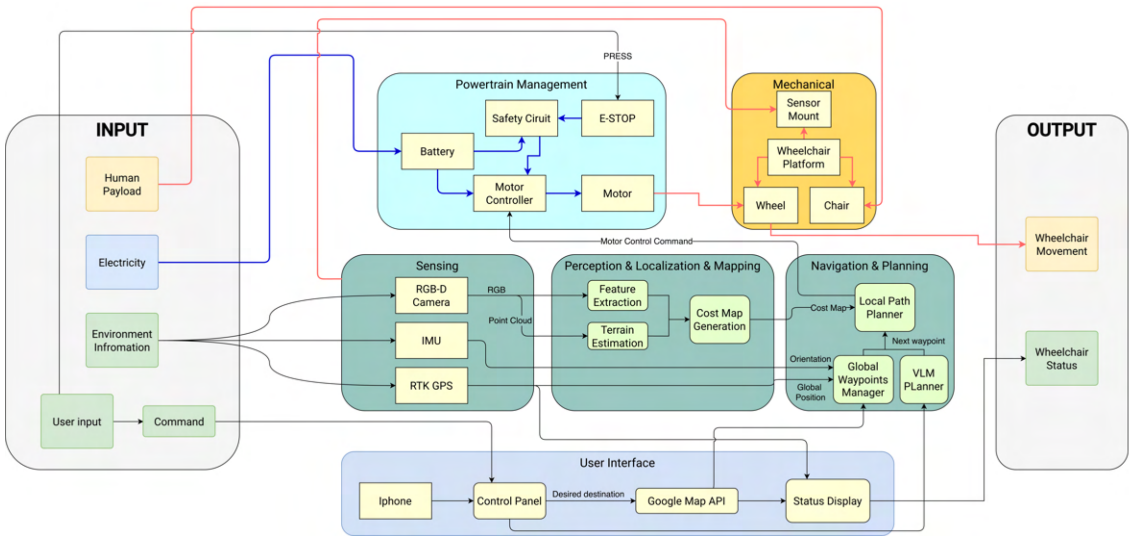


Figure 3: Cyberphysical Architecture

This design is organized into six functional domains: User Interface; Sensing; Perception, Localization & Mapping; Navigation & Planning; Powertrain Management; and Mechanical. Each domain interacts with adjacent modules through well-defined interfaces, ensuring the seamless exchange of sensor data, control commands, and safety signals between software and hardware components.

6.1 User Interface

The User Interface consists of a mobile application running on iOS devices (iPhone or iPad) and a local control panel node. Through the app, the user views a map integrated via the Google Maps API, which continuously updates the wheelchair's GPS location. Users can pin a destination directly on the map. Once the destination is transmitted to the global waypoint manager, the interface receives and displays the planned global path, allowing users to monitor trip status in real-time. For specific navigational preferences, users can trigger the Vision-Language Model (VLM) planner via the control panel. The VLM planner generates a proposed path overlaid on the First-Person View (FPV) image for user confirmation.

6.2 Sensing

The Sensing module acquires raw environmental data, applying timestamps for synchronization. An RGB-D camera captures color images and generates point clouds, providing both semantic context and depth measurements. Simultaneously, an Inertial Measurement Unit (IMU) provides orientation, angular velocity, and linear acceleration data to support dead-reckoning. To achieve centimeter-level positional accuracy, an RTK GPS receiver applies real-time corrections via an NTRIP stream. All sensor outputs are published as ROS 2 topics for downstream processing.

6.3 Perception, Localization & Mapping

RGB images are processed by a feature extraction pipeline to derive semantic features, while point clouds undergo a terrain estimation pipeline to extract geometric features. The point clouds also serve as a projection medium for the RGB images. Visual Foundation Model (VFM) features are projected into a Bird's Eye View (BEV) via point cloud projection. Subsequently, the BEV semantic and geometric features are concatenated and fed into a cost map generation model, the output of which serves as the input for the planning module.

6.4 Navigation & Planning

The Navigation and Planning module receives a user-specified destination and generates a global path based on a pre-defined wheelchair-accessible map. Utilizing the wheelchair's current location and orientation, the module transforms global waypoints into the robot's local frame. Alternatively, if the user engages the VLM planner, the system queries the VLM using an FPV image and the user's specific command to generate a path within the robot's local frame. Using these local waypoints, the local planner determines the optimal trajectory based on the generated cost map and transmits velocity commands to the motor controller.

6.5 Powertrain Management

Powertrain Management encompasses the battery pack, motor controller, electric motors, and safety circuit. The Battery Management System (BMS) monitors voltage, current, and temperature to protect against overloads and thermal events. Velocity commands are received via a CAN link, driving the motor controller to modulate motor torque. An Emergency Stop (E-STOP) button interfaces directly with the safety circuit; when activated, the circuit opens, causing the motor controller to immediately sever power to the motors.

6.6 Mechanical

This module comprises a robust wheelchair platform equipped with secure sensor mounts designed to mitigate vibrations. To ensure user comfort and safety, the assembly includes a seat featuring armrests and a seatbelt. Additionally, a footrest is incorporated for user support, and the wheels are engineered to withstand diverse terrain conditions.

7 System Description and Evaluation

7.1 Overall System Depiction

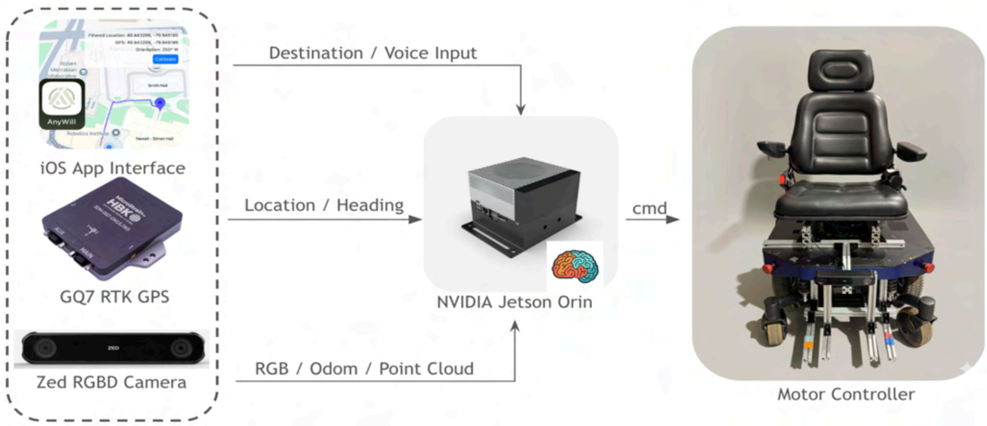


Figure 4: Overall System Depiction

The overall system architecture is depicted in Figure 4. We utilize a ZED-X camera and a 3DM GQ7 dual-antenna GPS to capture environmental data. An iOS application serves as the user interface for setting destinations, transmitting sensor data, and sending high-level commands to the onboard computing module, a Jetson AGX Orin. The Jetson AGX Orin operates on ROS2 Humble, providing the necessary computational power for the robotics software stack, including perception and planning modules. The planner generates velocity commands via a Copley-CAN controller, which subsequently publishes CAN BUS messages to the motor controller to actuate the wheelchair.

7.2 Subsystem Descriptions

7.2.1 Mechanical

The mechanical subsystem addresses two primary objectives: providing a stable mount for the sensors and the computing unit, and accommodating the human payload while ensuring user comfort and safety.

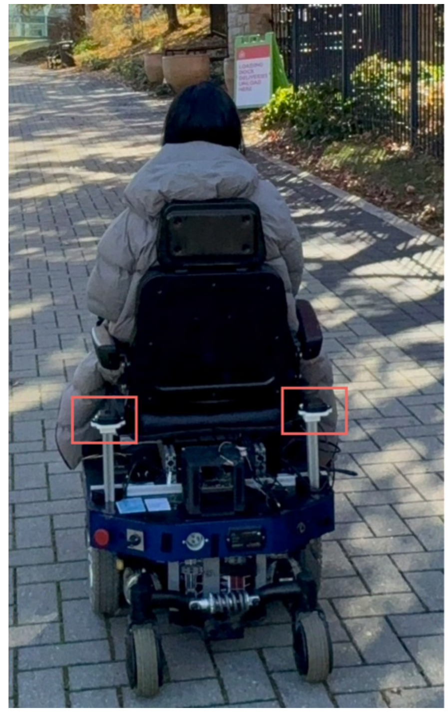
To achieve the first objective, the camera and the Orin are securely attached to the wheelchair platform. In the current iteration, the camera placement was adjusted to accommodate the new footrest design, ensuring the user does not obstruct the camera's field of view. We utilized a T-slot extension to mount the camera, providing a clear visual path when the user is seated, as illustrated in Figure 5a. Furthermore, two T-slots were employed to mount the dual-antenna system, minimizing interference from the seat and the metal chassis of the wheelchair, as shown in Figure 5b.

For the seating assembly, we selected a high-back suspension seat featuring thick cushioning, a built-in seat belt, and armrests. This seat is mounted on two aluminum extrusion rails, a design that effectively transfers load to the platform and supports a weight capacity of up to 200 pounds.

7.2.2 Powertrain

The powertrain subsystem encompasses the electrical hardware, including a battery for the motors and motor controller (Figure 6), as well as an E-STOP system for emergency intervention. A 12V

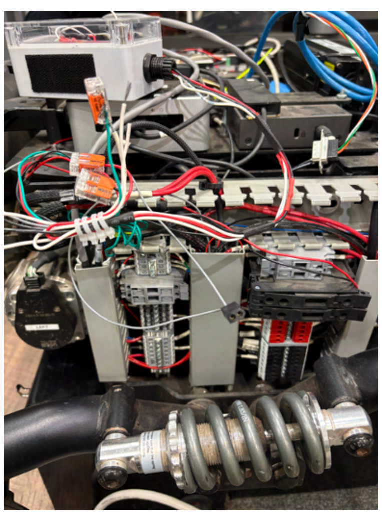
(a) Footrest and camera mount



(b) Dual-antenna mounting

Figure 5: Sensor mounting configurations

LiFePO4 battery powers the entire system. The motors and motor controller are securely installed and wired, with three emergency stop buttons strategically positioned on three sides of the wheelchair for accessibility. Currently, the maximum speed is governed at 1 m/s to maintain a balance between user comfort and operational safety.



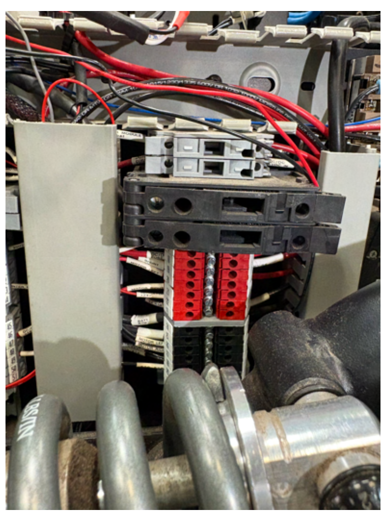


Figure 6: Powertrain system components

7.2.3 Sensing

The sensing subsystem is equipped with a ZED-X camera and a 3DM GQ7 RTK GPS. The ZED-X is a high-fidelity RGB-D camera offering extensive functionality via the ZED SDK. We utilize the RGB images and point clouds generated by the camera, operating at approximately 10 Hz. Additionally, the ZED SDK provides visual odometry, which is employed to estimate the wheelchair's pose.

For global localization, we selected the 3DM GQ7 dual-antenna GPS. This unit determines current coordinates and utilizes a dual-antenna configuration to ascertain the wheelchair's orientation. By integrating IMU fusion, we obtain a heading that is both precise and responsive. Regarding RTK corrections, we utilize the AirLab RTK base station. While we initially employed rtk2go.com as a host, network instability prompted the implementation of a custom RTCM streamer. This streamer transmits RTCM messages directly from AirLab via WebSockets within the CMU network, ensuring reliable data delivery to the wheelchair.

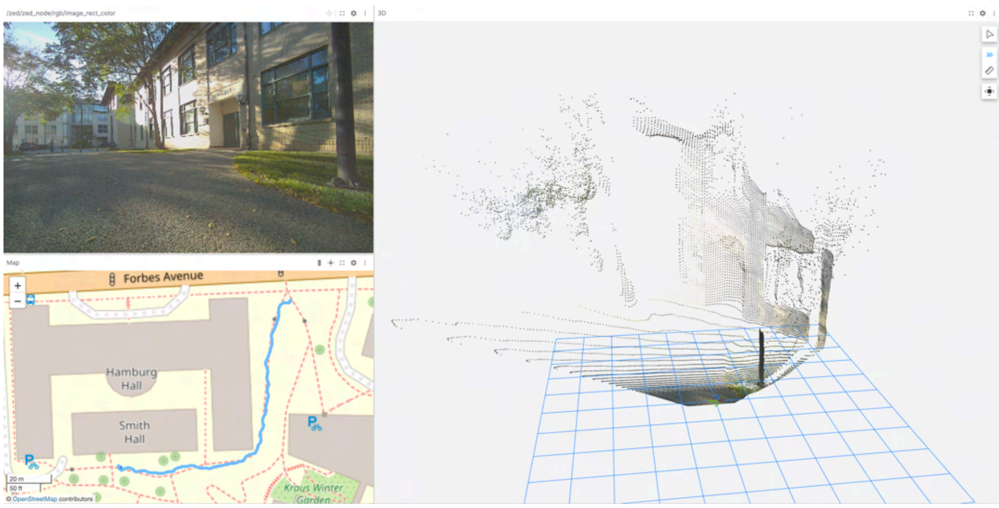


Figure 7: Sensor data visualization

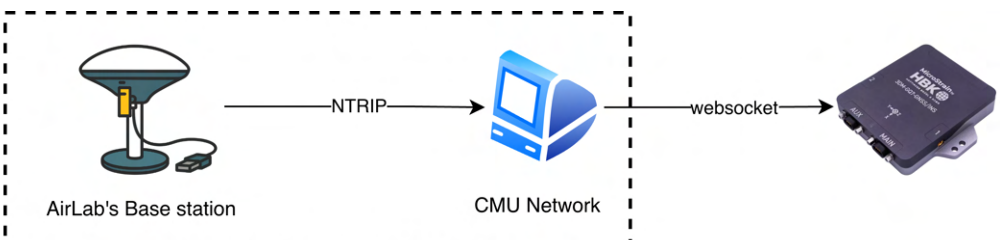


Figure 8: RTCM streamer data flow

7.2.4 Perception

In the perception stack, we leverage First-Person View (FPV) images and point clouds to generate a traversability cost map. As shown in Figure 9, point clouds are used for terrain estimation. Simultaneously, a Visual Foundation Model (VFM) encoder extracts semantic features from the FPV images. While our system supports using DINOv2 [3], SAM2 [4], and NACLIP [5] as the VFM encoder, our

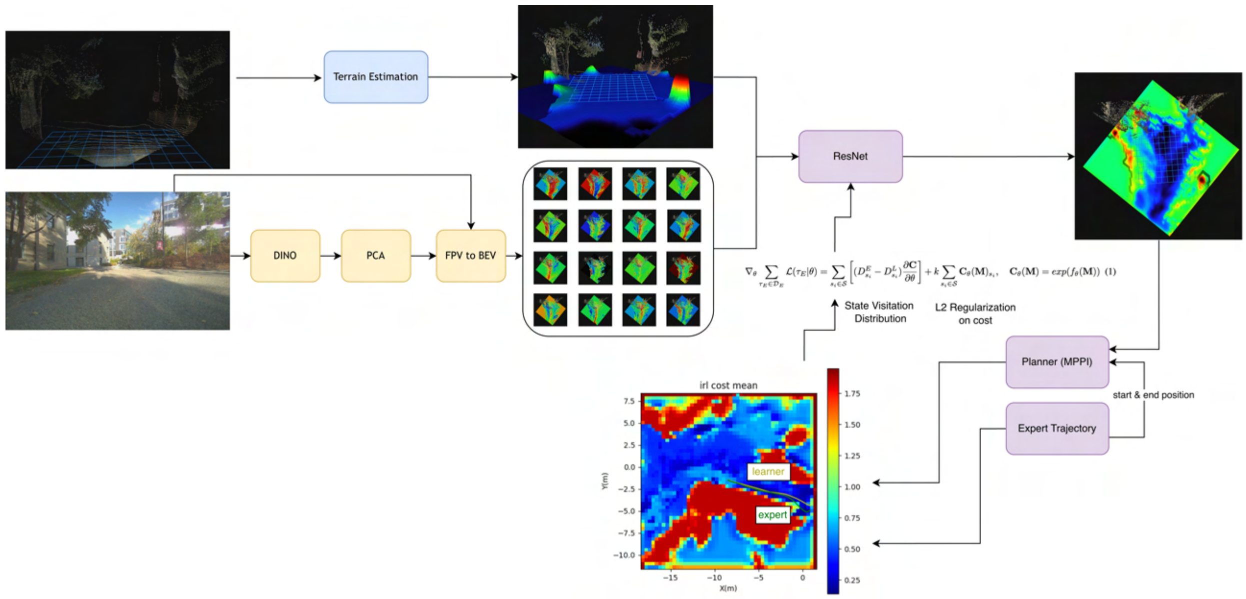


Figure 9: Perception stack architecture

final system implementation used RADIO [6] for its efficiency. Due to computational constraints, Principal Component Analysis (PCA) [7] is applied to reduce the dimensionality of these semantic features while retaining critical information. The point cloud is projected onto the FPV space using camera intrinsics, and semantic features are remapped to a Bird's Eye View (BEV) space. Both geometric and semantic features are then fed into a model to generate a BEV cost map. To train this model, we collected three hours of expert teleoperation data and performed Maximum-Entropy Inverse Reinforcement Learning (MaxEnt-IRL) [8]. The objective is to learn human preferences and generate a cost map that enables the planner to mimic expert demonstrations.

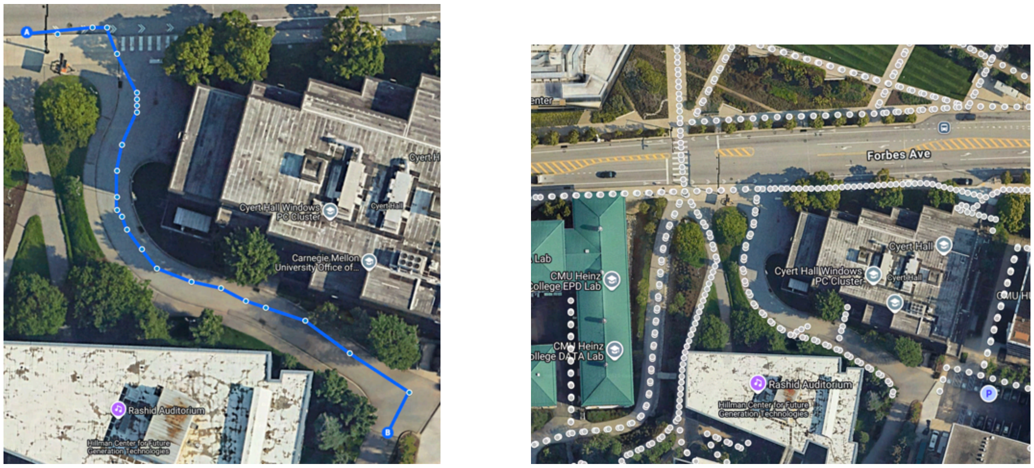
7.2.5 Planning

The planning subsystem utilizes a multi-stage approach. When a destination is set, the system first employs the A* algorithm [9] to generate a global path based on a pre-defined wheelchair-accessible map. This map is constructed using data from OpenStreetMap (OSM) [10]. OSM was selected over Google Maps due to its superior customizability and more detailed pedestrian layers, as illustrated in Figure 10. The resulting wheelchair-accessible map of the CMU campus is shown in Figure 11.

Once the global path is determined, global waypoints are transformed into the robot's local frame based on its current location and orientation. Waypoints are cleared if the distance to the wheelchair is less than 4 meters; this threshold accounts for potential GPS localization noise.

Additionally, we implemented a Vision-Language Model (VLM) planner to accommodate specific user preferences. Standard planners typically center the wheelchair on the sidewalk; however, a user may prefer to stay to the right. To address this, FPV images and user preferences are fed into a VLM. Our implementation enables using either OpenAI's GPT-4o [11] or Google's Gemini-2.5 [12], depending on the user's preference. The model leverages its reasoning capabilities to plan a path in the image space (Figure 12), which is then mapped to the robot frame using camera intrinsics.

While the VLM planner is powerful, it introduces significant latency (15-20 seconds per query), which limits real-time applicability. Consequently, GPS waypoints serve as the default navigation



(a) Google Maps waypoints **(b) OSM walkable areas**

Figure 10: Comparison between Google Maps and OSM data

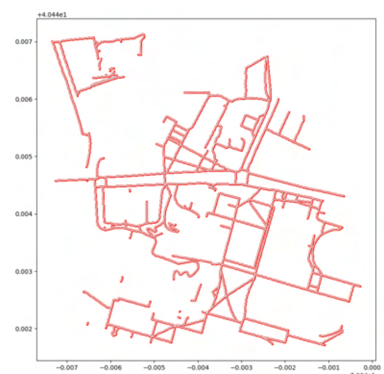


Figure 11: Wheelchair-accessible map of the campus

Figure 12: Direct FPV planning based on goal location

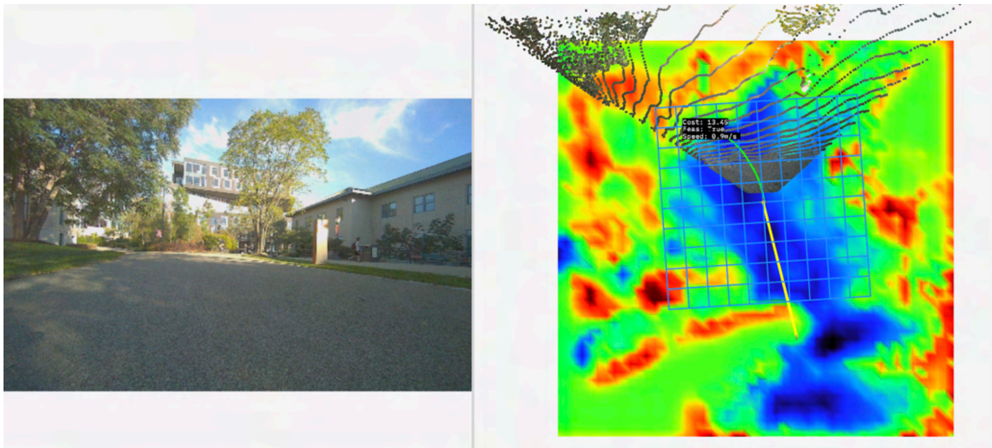
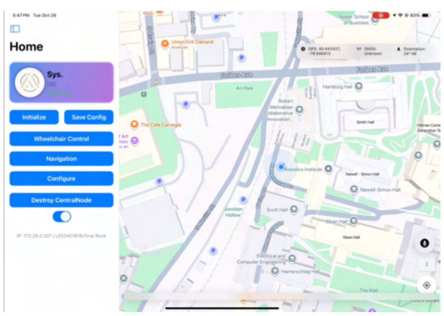
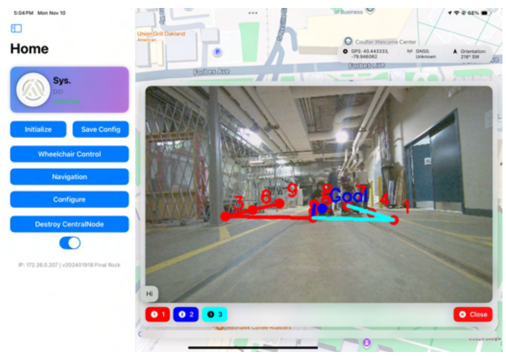


Figure 13: MPPI planning visualization



(a) General interface



(b) VLM interface integration

Figure 14: User Interface

method, with the VLM planner activated only upon specific user request via the UI.

Following the generation of local waypoints, the cost map generated in Section 7.2.4 is used for local planning. We employ Model Predictive Path Integral (MPPI) [13] control, for which we developed a custom action library and performed extensive parameter tuning. The system replans every 0.5 seconds to select the lowest-cost path to the next waypoint (Figure 13). If a valid solution is found, velocity commands are sent to the motor controller for execution.

7.2.6 User Interface

The system includes an iOS application for user interaction, as depicted in Figure 14a. Users can set destinations, view travel metrics, or input specific preferences. Communication with the wheelchair is facilitated through the DDS layer, enabling direct integration with the ROS2 interface. When the user triggers the VLM planner, the system waits for the VLM to generate a plan, which is then visualized on the UI. The user can review, accept, or reject the VLM path, or request a replan, as shown in Figure 14b.

7.3 Modeling, Analysis, Testing

During system design and implementation, we performed various modeling, analysis, and testing for improving the performance of the overall system. We include several important analysis and testing in this section that enabled our system's overall success.

7.3.1 Power Output for AGX Orin

While performing field tests with our full autonomy system, we noticed that sometimes our AGX Orin suddenly loses power and shuts down while the wheelchair is moving. Since this was a new issue encountered when we just integrated the latest autonomy stack, we first assumed it was a software issue. However, in our unit tests and full-system tests indoors, we never observed such an issue, and the system-level error logs right before the shutdown do not offer any informative suggestions to the root of the problem.

With further field testing and analysis, we identified the problem as the lack of sufficient power provided to the AGX Orin by our battery when the wheelchair climbs a slope. In order to improve the efficiency of our autonomy system, we enabled the maximum power consumption on AGX Orin, which requires at least 60W of power input stably, whereas the motors consume too much energy during the initial acceleration when climbing upslope such that this power requirement is interrupted to the AGX Orin. To ensure system robustness, we added a secondary power source for the electronics onboard to isolate their power consumptions and that of motor controls.

7.3.2 Camera Placement

Our initial system placed the camera in the front of the wheelchair platform, assuming an unobstructed field-of-view by the camera. However, this is no longer the case when we integrated the wheelchair seat to the system, as a wheelchair user's natural sitting position would have their legs obstruct most of the field-of-view of the camera. In order to address this problem, we performed an analysis and unit tests on the placement of the camera.

Initial proposals include placing the camera on top of the seat, to the left side of the wheelchair platform to unobstruct the view, and placing the stereo camera vertically instead of horizontally. However, further analysis and unit tests prove them unfeasible: placing camera on top of the seat makes the camera too far from the ground view that its depth point clouds no longer capture the poles and other minor safety hazards close to the road; placing it to the left side of the wheelchair will make the camera stick out of the side of the platform, violating the requirement of a wheelchair to safely traverse through regular doors as it increases the overall width of the wheelchair to a larger width than typical doors; placing the camera vertically was a good option, but with more unit tests we observed that the depth estimation from its stereo input is biased towards horizontal placement, and vertical placement negatively impacts the accuracy of depth estimation.

We finally settled on a packaged solution of an extension mount of the camera that sticks out to the front of the user's leg placements, and designed our footrests according to the specification of our camera mount to enable the user to comfortably sit on our wheelchair while not interfering with our camera's sensing.

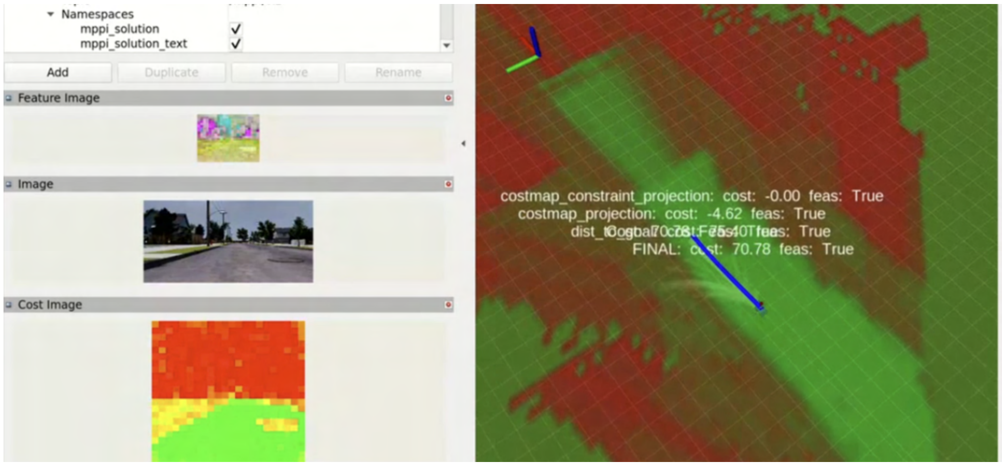


Figure 15: Development of First-Person-View Costmap in IsaacSim Neighborhood.

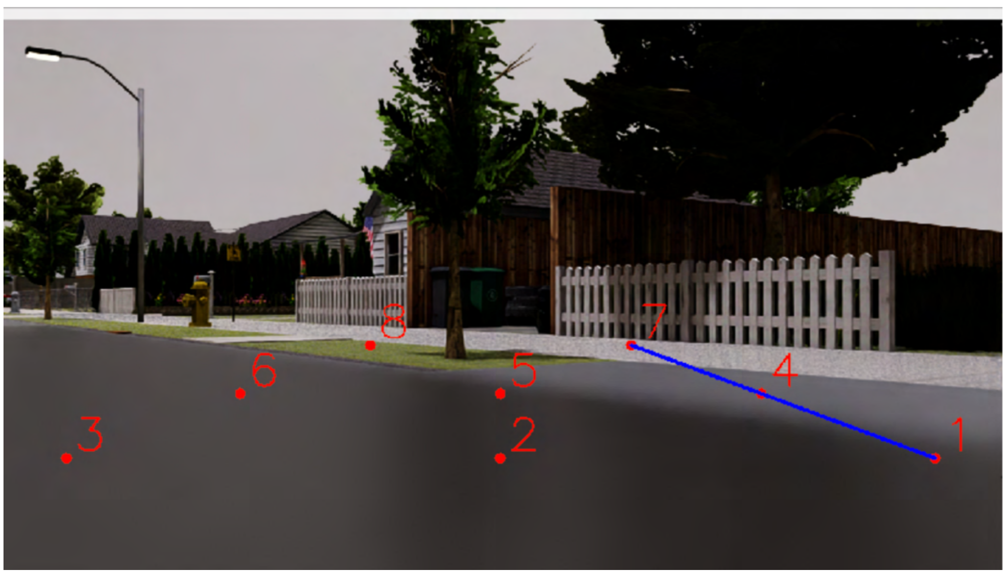


Figure 16: Development of VLM planner in IsaacSim Neighborhood.

7.3.3 Autonomy Testing in IsaacSim

Before integration and field testing on our physical wheelchair, we performed unit tests of the autonomy stack in an IsaacSim environment of a neighborhood street. We modeled the wheelchair as a TurtleBot, a two-wheeled mobile robot, which has the same principle dynamics as our two-wheel-controlled wheelchair.

To simplify the alternations between IsaacSim testing setup and physical wheelchair stack, we implemented mode selection configurations in our autonomy stack that switches from sim mode and real-world mode, with the key difference being the specific ROS2 topics used for different componenet of autonomy, and the expectations in sensor topic ages. Within the IsaacSim neighborhood environment, we performed most of our initial development and integration of the components of the autonomy stack, including perception, costmap learning, planning, controls, and VLM planner, as shown in Figure 15 and Figure 16.

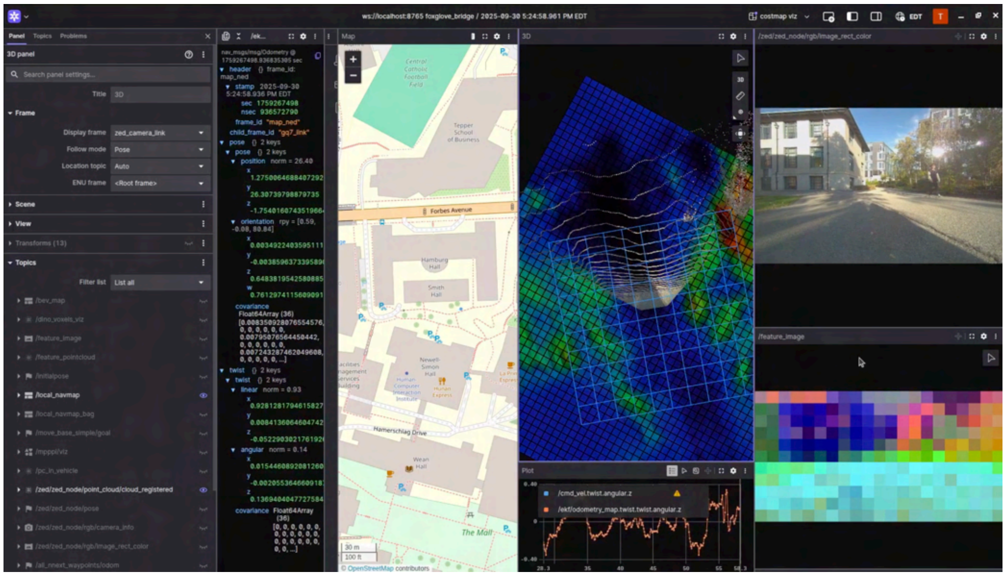


Figure 17: Offline autonomy testing with ROSBag playback topics.

7.3.4 Autonomy Testing with ROSBag Playback

In addition to using IsaacSim for initial development, integration, and testing, we used recorded ROSBags from our field tests to further develop the autonomy stack using captured sensor inputs. As our system becomes more mature and ready for real-world execution, it becomes more efficient to perform autonomy testing with pre-recorded topics of inputs, including sensor inputs such as RGB image, depth point clouds, GPS coordinates, and command inputs such as specified goal locations. By only subscribing to the topics desired, we can extract arbitrary inputs we need from all the topics we captured and run a modified autonomy stack to test its performance on real-world data, shown in Figure 17.

7.4 FVD Performance Evaluation

The Fall Validation Demonstration (FVD) was conducted along a real outdoor sidewalk environment around Schenley park. The route started from the botanical garden entrance at GPS coordinate (40.4395, -79.9468) and terminated at the destination point (40.4411, -79.9476), with a total travel distance of approximately 450 meters. The route includes three dedicated testing scenarios: (1) planned rerouting around static pedestrians for human avoidance evaluation, (2) safe stopping behavior in front of dynamic pedestrians, and (3) a VLM-integrated semantic replanning scenario triggered by user commands.

The objective of this demonstration was to validate the full-stack autonomy of the wheelchair system, including perception, localization, motion planning, semantic replanning, control, and user interface integration in a real-world outdoor environment.

7.4.1 Demonstration Objectives

The FVD autonomous navigation demo was designed to demonstrate the following core system capabilities:

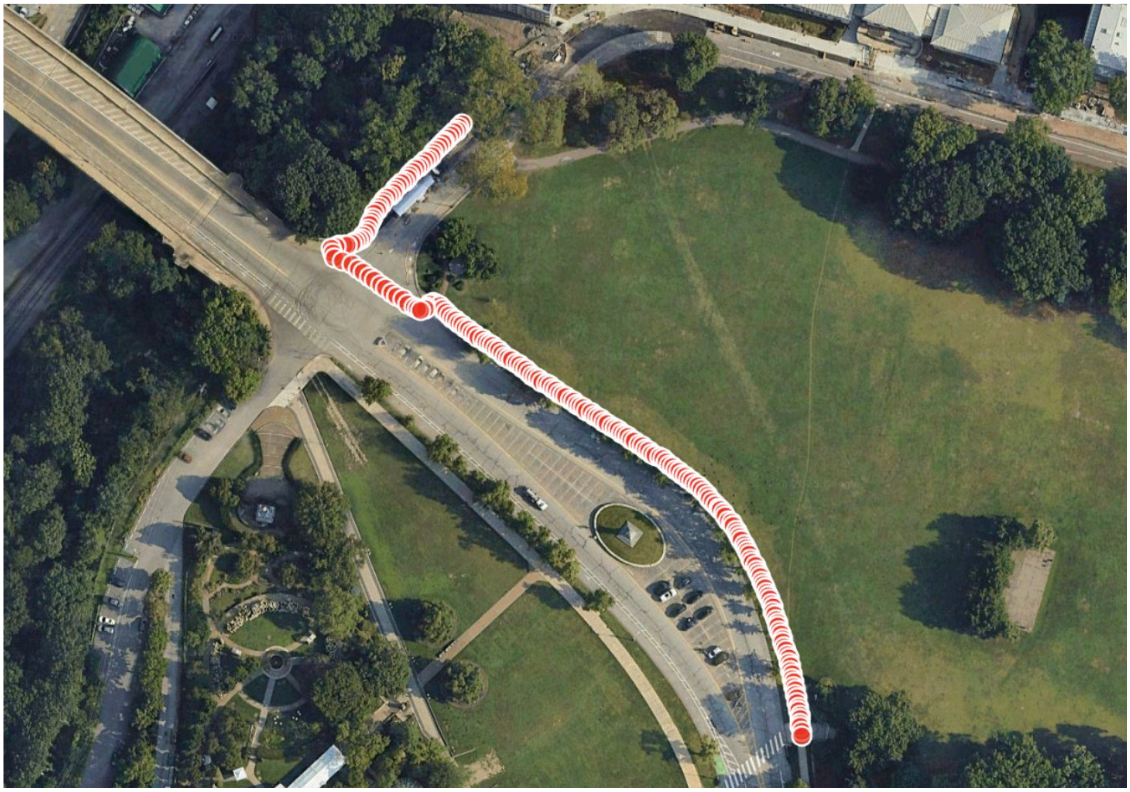


Figure 18: Outdoor test route for the Fall Validation Demonstration (FVD) along Schenley Drive.

- Fully autonomous outdoor navigation along a predefined sidewalk route
- Planned avoidance behavior around static pedestrians
- Safe stopping behavior in response to dynamic pedestrians
- Robust on-curb and off-curb navigation
- High-level semantic replanning through VLM integration

7.4.2 Success Criteria

The success of the FVD demonstration was defined based on the following evaluation criteria:

- Completion of the full route within 8 minutes
- Successful bypass of static pedestrians
- No collision with moving pedestrians through safe stopping behavior
- Successful avoidance of all encountered obstacles
- Human intervention (E-STOP or teleoperation) fewer than 2 times
- Correct use of sidewalk entrances for both entering and leaving the sidewalk
- Reaching the destination within 5 meters of the target location

7.4.3 Performance Results

The FVD run successfully satisfied all verification criteria, with the total execution time slightly exceeding the 8-minute target. The wheelchair reached the final destination with a positional accuracy of 1.2 meters, significantly better than the 5-meter requirement. The full navigation sequence (Steps 1–9) was completed in approximately 9 minutes. The system demonstrated correct detection and usage of sidewalk entrances and exits.

Both static and dynamic human obstacles were successfully avoided multiple times during the run. The VLM-based high-level planner successfully generated an alternative route that allowed the wheelchair to bypass a cup placed in the middle of the sidewalk.

Only one brief teleoperation intervention was required due to temporary GNSS degradation. In addition to the predefined test route, the system also successfully completed several randomly selected reverse-direction navigation trials on the same sidewalk.

7.5 Strong / Weak points

Our progress throughout the semester culminated in successful demonstrations during both FVD and FVD Encore. While the system demonstrated strong performance across core autonomy components, several limitations were also identified that indicate clear directions for future refinement. The key strengths and weaknesses of the current system are summarized below.

7.5.1 Strengths

- **Robust End-to-End Autonomous Navigation:** The system demonstrated reliable full-stack autonomy, including perception, localization, motion planning, and control, enabling consistent outdoor navigation along complex sidewalk routes.

- **Effective Static and Dynamic Human Interaction:** The wheelchair successfully planned around static pedestrians and executed safe stopping behavior in response to dynamic pedestrians multiple times, validating both planning and safety mechanisms.

- **Successful VLM-Based High-Level Replanning:**The VLM-integrated semantic planner successfully generated alternative routes to bypass obstacles based on user commands.

- **Reliable Sidewalk Entry, Exit, and Curb Traversal:** The system consistently executed off-curb and on-curb maneuvers, including multiple reverse-direction navigation trials beyond the predefined test route.

- **Strong System Reliability and Repeatability:** Across multiple test runs, the platform demonstrated stable system behavior with minimal human intervention, indicating solid software-hardware integration.

7.5.2 Weaknesses

- **Sensitivity to GNSS Degradation:** The system exhibited continued reliance on high-quality GNSS signals. Temporary GNSS degradation required brief teleoperation intervention, indicating the need for stronger multi-sensor localization fusion.

- **Traction and Low-Speed Actuation Limitations:** Minor wheel slip and brief instability were observed under certain terrain conditions. At low speeds, the motion control was occasionally not smooth due to uneven force distribution between the left and right wheels, which sometimes caused slight directional drift. These phenomena indicate that both low-speed control tuning and the mechanical drivetrain design can be further improved.
- **Limited Robustness to Adverse Environmental Conditions:** Localization and perception performance remain sensitive to adverse environmental conditions. Reduced lighting conditions negatively affect visual perception reliability, while GNSS signals are vulnerable to partial occlusion in cluttered outdoor environments. In addition, the current system lacks waterproofing, preventing safe operation and testing under rainy conditions. Besides, thermal instability was observed in previous summer tests, where high ambient temperatures led to system overheating and intermittent performance degradation. These factors limit overall system stability under extreme environmental conditions and indicate the need for improved environmental protection and thermal management.
- **Limitations in VLM-Based Semantic Replanning:** The precision of VLM-generated intermediate waypoints is affected by depth estimation accuracy, leading to relatively coarse waypoint placement at longer distances. Furthermore, the current VLM query latency remains on the order of tens of seconds per iteration, which constrains its responsiveness in time-critical scenarios. Further optimization of both depth estimation and inference efficiency is required to better support real-world deployment.
- **Curb Detection Sensitivity to Road Surface Variations:** The curb detection performance is influenced by variations in road surface conditions. Uneven sidewalk tiles and small height discontinuities may occasionally introduce ambiguity in curb perception. Since curb detection accuracy is closely coupled with local planning decisions for curb ascent and descent, such uncertainty can affect the smoothness and stability of curb traversal. Further improving the robustness of curb detection under non-ideal surface conditions remains an important direction for future refinement.
- **Scalability to More Densely Populated Environments:** Although human avoidance performed well in the tested scenarios, denser pedestrian traffic and faster dynamic agents may require more aggressive prediction and behavior modeling.

8 Project Management

8.1 Schedule

This section summarizes our Fall project schedule, compares planned vs. adjusted timelines, and evaluates our scheduling process.

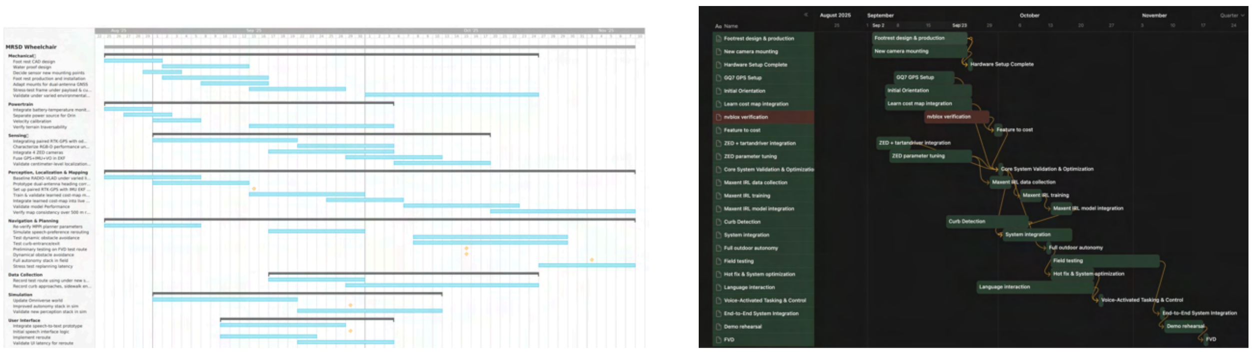
8.1.1 Planned and Adjusted Schedule

Key milestones and bi-weekly tasks are summarized in Table 9. External milestones are in bold; internal tasks focused on integration, costmap improvements, and language interface testing.

Table 9: Key System Development Milestones (Bold = external)

Date	Milestone
September 24	Milestone 1 (M1) – Integrate RTK-GPS + IMU; add speech interface
October 29	Milestone 2 (M2) – Validate autonomy stack; demo obstacle avoidance
November 17	**Fall Validation Demonstration (FVD) – Full system demo**

Figure 19 shows the original planned schedule (left) and the adjusted final schedule (right), allowing a compact comparison.



(a) Planned Schedule

(b) Adjusted Schedule

Figure 19: Comparison of Planned vs. Adjusted Fall Project Schedule

8.1.2 Evaluation of Scheduling Process

The adjusted schedule was followed effectively. Subsystems progressed on time, though system integration slipped one week due to additional features and MaxEnt IRL updates. Recovery strategies included separate outdoor tests, offline data integration, and removal of unnecessary tasks. These measures ensured completion before FVD.

Successes:

- Maintained high efficiency by prioritizing essential tasks.
- Rapidly adapted to delays and implemented recovery strategies.
- All demonstration deliverables achieved through realistic scheduling.

Failures and Areas for Improvement:

- Insufficient buffer for outdoor testing; unexpected issues arose during FVD.
- Underestimated system integration difficulty, leading to a one-week delay.

8.2 Budget

Table 10 presents the finalized parts list with quantities, descriptions, and costs for all purchased components. The total financial cost of the project is $2,309.59, covering electronics, computation hardware, mechanical seat components, wiring accessories, and power supplies. High-value equipment was borrowed from AirLab, including four ZED X Stereo Cameras (estimated $624 each) and a wheelchair platform (estimated $5,000). Borrowed items are accounted for in risk/value assessment but excluded from the total financial cost.

Table 10: Key Components Purchased for MRSD Project Course

Qty	Part No.	Part Name	Unit Price	Total Price	Description
1	GPS1	Ublox ZED-F9P GNSS	$314.95	$314.95	High-precision GNSS for wheelchair autonomy
1	GPS2	Ublox ZED-F9P GNSS	$314.95	$314.95	Second GNSS module for redundancy
1	DT5	Jetson Orin Nano	$249.00	$249.00	Onboard autonomy computation
1	DT6	ZED Mono Capture Card	$139.00	$139.00	Camera interface for perception system
1	DT1	Portable Monitor	$239.00	$239.00	Operator interface for testing
1	GPS3	USB Cable Kit for AHRS/IMU	$152.78	$152.78	GNSS connection and data acquisition
1	CH1	Universal Lawn Mower Seat	$123.99	$123.99	Wheelchair seat assembly
-	-	Remaining components	-	$775.92	Cables, connectors, relays, power supplies, and integration hardware
			Total	**$2,309.59**	

8.2.1 Budget Analysis

The largest expenditures were high-precision GNSS modules, the Jetson Orin Nano, ZED Mono capture card, portable monitor, and USB cable kit. These were essential for autonomy, sensing, and operator usability. Mid-range items, such as the wheelchair seat and networking equipment, supported functionality and comfort. Low-cost components (cables, relays, connectors) facilitated prototyping and system integration.

8.2.2 Evaluating the Budgeting Process

Successes:

- Strong cost control: spent less than 50% of allocated $5,000 budget while meeting all functional requirements.
- Prioritized high-cost items for core autonomy; low-cost components supported integration and testing.
- Timely ordering of long-lead items minimized delays.
- Clear and organized documentation simplified tracking and reconciliation.

Failures and Areas for Improvement:

- Some components (cables, brackets) were ordered reactively rather than pre-planned.
- Underestimated small accessories, resulting in multiple minor orders.
- High-cost items added late (e.g., second GNSS module) could have benefited from earlier planning.

8.3 Risk Management

Throughout the MRSD project course, we actively managed risks to ensure safety, system reliability, and timely completion of milestones. Table 11 summarizes the most critical risks, their priority, status, responsible team member, and category. We selected high-priority risks and representative medium-priority risks that best illustrate the challenges faced in mechanical, software, UI, and hardware domains.

Table 11: Key Risks for MRSD Project Course

ID	Description	Status	Likelihood	Severity	Owner	Category
R5	Sim-to-real gap	Done	High	High	ChaoI Tuan	Software
R15	Over drifting of GPS	Done	High	High	Chiawen	Software, UI
R16	Hardware components overheating	Done	High	High	Yu-Hsin Chan	Hardware, Software
R21	Current baseline model does not detect curbs or curb cuts	Done	High	High	Chiawen	Software
R27	Current autonomy stack may not recognize sidewalk edge	Done	High	High	Chiawen	Software
R33	Wheelchair might run onto road if it doesn't recognize sidewalk edge during testing	Done	High	High	Yu-Hsin Chan	Software
R28	Model trained on ATV off-road data is sensitive to trees but fails to assign high cost to humans	Done	High	High	ChaoI Tuan	Software
R1	Remote E-STOP design fail	Done	Medium	Medium	Yu-Hsin Chan	Mechanical

ID	Description	Status	Likelihood	Severity	Owner	Category
R11	Remote E-STOP failed in outdoor testing	Done	Medium	Medium	Yu-Hsin Chan	Mechanical
R13	Sudden rains during outdoor testing damages Orin and circuits	Done	Low	High	Yu-Hsin Chan	Mechanical
R20	Precipitation	Done	Low	High	Sonic	Hardware
R31	UI app interaction with VLM planning subsystem encounter timing issues	Done	Medium	Medium	Chiawen	UI

8.3.1 Risk Summary and Mitigation:

The most critical risks involved safety, autonomy, and hardware reliability. High-priority risks included GPS over-drifting (R15), hardware overheating (R16), sidewalk edge detection failures (R21, R27, R33), and potential misbehavior of the autonomy stack (R28, R5). These risks were mitigated through iterative simulation-to-real testing, redundancy in sensing (dual GNSS modules), robust power management, and conservative safety protocols during outdoor trials.

Medium-priority risks such as E-STOP design and outdoor robustness (R1, R11, R13) were addressed with prototype validation and modular testing. Timing conflicts in UI interactions (R31) were managed by phased software integration and continuous testing. Low-priority environmental risks (R20) were monitored, and contingency measures—like waterproof enclosures for sensitive electronics—were applied.

Successes:

- High-priority risks were identified early, allowing sufficient time for mitigation before critical testing phases.
- Clear assignment of ownership ensured accountability and efficient responses.
- Iterative testing, both in simulation and real-world trials, minimized unsafe maneuvers and hardware damage.
- Backup systems, including remote E-STOP and dual GNSS modules, improved overall safety and reliability.
- Systematic documentation enabled effective tracking of mechanical, software, and UI risks.

Failures and Areas for Improvement:

- Environmental risks such as sudden rain were only partially anticipated, causing minor delays and additional protective measures.
- Hardware constraints, particularly limited battery life for the Orin and Wi-Fi router, restricted some test sessions and required careful planning.
- Certain software integration issues, such as timing conflicts between the UI and VLM planner, required multiple iterations to fully resolve.

- Some low-priority risks (e.g., minor perception issues from fallen leaves) were underestimated, causing small disruptions during testing.

Overall, our risk management process effectively prevented critical failures, maintained safety, and allowed smooth progress toward project milestones. Early identification, assigned ownership, iterative testing, and adaptive mitigation strategies contributed to a robust approach across all subsystems and environmental conditions.

9 Conclusions

9.1 Lessons Learned

9.1.1 Early System Integration Is Critical.

One of the most important lessons learned is the necessity of performing system integration as early as possible. In our case, several sensing and perception components were integrated at a later stage, at which point a hardware failure in the ZED camera was discovered. This late discovery significantly affected the overall system performance and delayed debugging. This experience highlights the importance of early end-to-end integration to expose hidden hardware and interface issues before they become critical bottlenecks.

9.1.2 Well-Maintained Documentation Aligns the Team and Accelerates Collaboration.

Comprehensive and continuously updated documentation is essential for keeping all team members on the same page. Clear documentation enables more effective technical discussions, reduces miscommunication between subsystems, and creates greater room for collaborative problem solving. In our project, improved documentation directly enhanced transparency, cross-team coordination, and overall development efficiency.

9.1.3 Multiple Dry Runs Under Diverse Conditions Reveal System

Conducting multiple full-system dry runs at different times, weather conditions, and physical locations proved to be essential for understanding system capabilities and failure modes. These repeated tests allowed the team to identify performance limitations, environmental sensitivities, and rare edge cases at an early stage. For example, higher-resolution perception and additional point cloud filtering for obstacle detection were introduced based on observed system behavior to improve overall stability. This practice significantly reduced the risk of unexpected failures during final demonstrations and strengthened overall system reliability.

9.2 Future Work

9.2.1 Robustifying the System

If this project were to be used as the basis for a startup, the first priority would be to transform the current prototype into a robust and reliable platform that can operate safely in a wide range of real-world conditions. A key step would be addressing the system's dependence on high-quality GNSS signals. We would develop a tighter multi-sensor fusion stack that combines RTK GNSS with visual-inertial odometry, wheel odometry, and map-based localization so that the wheelchair can maintain stable performance even under GNSS degradation or occlusion.

In parallel, we would improve low-speed control and traction. This would involve revisiting motor controller tuning, closed-loop feedback gains, and possibly the mechanical drivetrain to minimize wheel slip, reduce asymmetry between the left and right wheels, and ensure smoother, more predictable motion at low speeds and on sloped or uneven terrain. The autonomy stack would also be extended to better handle adverse surface and lighting conditions. We would expand the training dataset for costmap learning to include nighttime operation, rain, snow, and cluttered or damaged sidewalks, and consider incorporating additional sensing modalities such as secondary cameras to increase robustness in visually challenging environments.

Finally, the hardware platform itself would be hardened for everyday use. Future iterations would feature improved waterproofing and dust protection, better cable management, and enhanced thermal management for the compute and power electronics. Together, these changes would move the system from a research prototype toward a dependable platform that can meet the safety, reliability, and maintainability expectations of real users and potential regulatory bodies.

9.2.2 Diverse User Testing

Once the technical foundations are more robust, the next major step would be systematic user testing with a diverse group of wheelchair users. The current evaluation has been limited in scale and diversity, and a startup path would require much stronger evidence of usability, safety, and user acceptance. We would seek partnerships with rehabilitation clinics, hospitals, assisted living facilities, and disability advocacy organizations to recruit participants spanning different ages, diagnoses, driving experience levels, and usage contexts (indoor, outdoor, campus, community).

Insights from this process would drive iterative improvements to the human-interactive components of the system. For example, the natural-language interface and visual feedback could be refined to better match users' mental models, and preference controls could be simplified into intuitive modes (e.g., "avoid bumps," "shortest time," "maximum comfort"). Overall, this co-design approach would ensure that future development is guided by real user needs rather than technical performance metrics.

9.2.3 Market Research

In parallel with technical and user-centered development, a startup based on this system would need a clear market strategy. The autonomous wheelchair and assistive mobility space includes a variety of products, ranging from high-end powered wheelchairs to retrofit autonomy kits and indoor-only navigation aids. We would begin with a structured market analysis to map this landscape, identify direct and indirect competitors, and understand where our system's capabilities provide a unique advantage.

This research would include segmenting potential customers into groups such as individual users living at home, long-term care facilities, rehabilitation hospitals, university and corporate campuses, and specialized mobility service providers. Based on the findings, we would decide on the most promising product direction: for example, a retrofit autonomy kit for existing powered wheelchairs, a fully integrated autonomous wheelchair platform, or a service model operated in partnership with institutions. We would then define a Minimum Viable Product that targets a specific, well-understood use case (such as autonomous mobility within a hospital or assisted living campus) and plan small pilot deployments with selected partners. Data gathered from these pilots on reliability, maintenance costs, user satisfaction, and clinical or operational impact would inform iterations of both the product and the business model, guiding the next phase of investment for an autonomous wheelchair startup.

References

[1] Caryne Torkia, Denise Reid, Nicol Korner-Bitensky, Dahlia Kairy, Paula W. Rushton, Louise Demers, and Philippe S. Archambault. Power wheelchair driving challenges in the community: a users' perspective. *Disability and Rehabilitation: Assistive Technology*, 10(3):211–215, 2014.

[2] Leon Salm, Lucas Schreff, Christian Benker, Rainer Abel, and Roy Müller. The effect of a cognitive dual-task on the control of wheelchair propulsion. *PLOS ONE*, 20(2):e0317504, 2025.

[3] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[4] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023.

[5] Sina Hajimiri, Ismail Ben Ayed, and Jose Dolz. Pay attention to your neighbours: Training-free open-vocabulary semantic segmentation. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5061–5071. IEEE, 2025.

[6] Mike Ranzinger, Greg Heinrich, Jan Kautz, and Pavlo Molchanov. Am-radio: Agglomerative vision foundation model – reduce all domains into one, 2024.

[7] Harold Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417–441, 1933.

[8] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, Anind K Dey, et al. Maximum entropy inverse reinforcement learning. In *Aaai*, volume 8, pages 1433–1438. Chicago, IL, USA, 2008.

[9] Peter E Hart, Nils J Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

[10] Mordechai Haklay and Patrick Weber. Openstreetmap: User-generated street maps. *IEEE Pervasive Computing*, 7(4):12–18, 2008.

[11] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[12] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

[13] Grady Williams, Austin Aldrich, and Evangelos A. Theodorou. Model predictive path integral control: From theory to parallel computation. *Journal of Guidance, Control, and Dynamics*, 40(2):344–357, 2017.