Boston University

ENG ME461: Senior Design

Cover Image: A Painting of Sisyphus by Vladmir Kush

Final Report

Team 183: D.U.N.G.

Advisor: Dr. Scott Bunch
Client: Dr. Kenneth D. Sebesta
Group: Arnav Singh, Lorenzo Barale, Maysarah Sukkar, Yousif Alhajji, Zachary Wu, Zaid Bhatti

Abstract:

Over the past year, the D.U.N.G. (Device Uplifting and Navigation Group) team developed an autonomous aircraft maneuvering system designed to safely and efficiently reposition aircraft inside and outside of hangars. Traditional aircraft handling is labor-intensive, imprecise, and prone to "hangar rash" - damage that may appear minor but can often be much more serious. To address this, the D.U.N.G. system uses three robots that use holonomic motorized casters, a cradle mechanism to secure and lift aircraft, and a sensor suite that allows for precise movement, localization, and obstacle avoidance. A web-based interface was also designed to allow users to easily schedule aircraft bookings and maintenance. Through extensive prototyping and multiple iterations across mechanical, electrical, and software subsystems, the team successfully built three demonstrator robots capable of lifting and maneuvering a scaled-down aircraft model. This project establishes a strong foundation for future senior design teams to build upon, enabling further improvements in safety, maneuverability, and user experience over existing commercial solutions.

Table of Contents:

List of Figures:

Section 2. Research, Benchmarking and Competitive Assessment

Section 5. Product Generation

Section 6. Product Evaluation

Section 7. Design Summary, Final Conclusions and Recommendations

List of Tables:

Section 3. Customer Requirements

Section 5. Product Generation

Section 6. Product Evaluation

Section 9. ABET Outcome

Executive Summary:

Current aviation practices reveal a critical need for more efficient methods of maneuvering aircraft in storage areas and hangars. Traditional repositioning often requires multiple personnel, intricate navigation around obstacles, and careful extraction of planes behind others, particularly in community hangars. These challenges stem from the aircraft's limited maneuverability and wide turning radii, making routine ground operations cumbersome, time-consuming, and potentially hazardous.

This project proposes a fully autonomous aircraft maneuvering system designed to streamline the repositioning process. Central to the solution is a semi-motorized caster wheel arrangement in an "X" formation, allowing holonomic motion. This configuration reduces both algorithm complexity and overall concept implementation for the robot's processing systems. The final products also use a ramp/cradle lifting system, which ensures compatibility across various tire sizes and avoids the added complexity of designing customized U-bar attachments.

Due to the autonomous nature of the project, a combination of sensors was planned to be used. However, after much deliberation the team decided to focus mostly on computer vision approaches and direct on-bot sensing options such as encoders and limit switches. On the software's front end, a web-based user interface built on the MERN stack was selected for its accessibility and user-friendliness. It includes secure login features, scheduling capabilities, and maintenance tracking options. Underlying this interface is an embedded system architecture that uses a Star communication topology, and robust path-planning through a modified Floyd-Warshall algorithm.

When compiled, the final product aims to outmatch its competition in terms of maneuverability and user experience. The team built a scale prototype to demonstrate system functions and while a real system's complexity and associated costs would likely be higher than its competitors, these are outweighed by D.U.N.G.'s safety and ease of use. With careful design iteration, the project could be scaled up and commercialized as a product given the prototype's success in maneuvering a scale model plane around a room.

Acknowledgements:

Team D.U.N.G. would like to extend their gratitude to the ME460/461 faculty team at Boston University for their support and advice throughout the course. Major thanks to Professor J. Scott Bunch for his consistent interest and effort during the process.

The team would like to thank the pilots at Mansfield Municipal Airport for their cooperation, kindness, and approachability during the interview stage of the process.

The team also thanks Professor Kenneth Sebesta for his guidance and encouragement as a client and as a project advisor.

Special thanks go to Principal Engineer Gabriel Hebert for his mentorship and advice throughout this project cycle and for arranging the site visit to an Amazon Robotics Facility.

Thank you to D.U.N.G. team member Yousif Alhajji for the work on the embedded systems of the project.

Thank you to D.U.N.G. team member Lorenzo Barale for the work on developing the front and back end of the web application and software design.

Thank you to D.U.N.G. team member Zaid Bhatti for handling the lifting and attachment portions of the project.

Thank you to D.U.N.G. team member Arnav Singh for working on the mobility aspect of the project.

Thank you to D.U.N.G. team member Maysarah Sukkar for the work on handling the controls portion of the project.

Thank you to D.U.N.G. team member Zachary Wu for the work on the motion planning portion of the project.

1. Introduction and Background:

The world of aviation is one brimming with marvels and adventure. In the modern age, pilots soar the skies in their metallic birds for a plethora of reasons. However, the challenge of getting an aircraft out of its storage space and into the air is one of precision, commitment, and endurance.

After speaking with various pilots and gathering their thoughts and experiences, it is clear that aircraft maneuverability is a significant issue as discussed in the customer requirements section. Specifically, during the preparation and storage of the aircraft, pilots are tasked with delicately moving in and out of their hangars and navigating around obstacles and other aircraft. Therefore, the wide turning circles of airplanes work against current users when trying to reposition their vehicles in their spaces to their needs.

Furthermore, it is difficult for some pilots to move aircraft without assistance. This issue is even more prevalent if the desired aircraft is stored behind another, such as in community hangars, where there may be a need to move a multitude of aircraft to reach the target.

In light of the aforementioned setbacks when it comes to aviation, the need for an autonomous aircraft maneuvering system has become apparent.

The team behind D.U.N.G. came together to face the challenge of developing an autonomous robotic system and address these issues. The project was initially proposed by a member of the faculty Professor Sebesta, who came up with the idea of an aircraft manipulation solution. However, some of the team met with the professor before officially taking on the project and proposed a re-scope of its requirements and deliverables. Once negotiations were over, the project description was adjusted to be more realistic and achievable within the one-year time frame that the students were given. The major changes involved the elimination of the optimal aircraft packing requirements and the reduction of 3D path planning considerations to 2D.

During the summer leading up to the fall 2024 semester, the team was put in contact with a principal engineer at Amazon Robotics, Gabriel “Gabe” Hebert, who became a mentor for the team in the coming months providing guidance and input on the engineering design process.

Toward the beginning of the fall semester, the team arranged a visit to Mansfield Municipal Airport where customer stories were heard and a more concrete problem statement was refined. At this visit, the students were able to live the experience of moving real planes around an airport and gain a better understanding of what pilots go through during the process as well as get a better idea of how the system could be improved.

In October 2024, the team, with Professor Sebesta, was invited to visit the Amazon Robotics facility in North Reading, MA. While at the site, the students were walked through the design process at Amazon Robotics and given a tour of the assembly line at the facility. While there, the team was given the opportunity to brainstorm ideas for the project with the guidance of Amazon Robotics engineers, the results of which were heavily influential to the final design.

Over the course of the fall semester, the team produced three progress reports each covering a given stage of the design process. Initially, a problem definition report was written to formalize and record the team's findings after speaking to potential customers and gathering user stories. This report set the foundation for how the project was going to be approached and what aspects of the problem the team had to prioritize when down-selecting concepts. The second report covered the concept generation stage of the process where the team segmented the project into its constituent parts and brainstormed ideas for each. The third progress report was about concept selection and compilation of the ideas into a single design. The results of these are briefly discussed in this report in Conceptual Design Summary section which aims to act as an overarching description of the team's operations during the fall semester.

Over the winter break, team DUNG worked on preliminary designs for the robots and planned their approach to building the system. The previous plans were revisited and analyzed to gauge whether they were feasible or if they had to be reworked to fit new constraints.

When the team returned to campus after the winter holiday, the product generation phase of the project was started. In this stage, an initial robot prototype was built and a progress report was submitted that encompassed the details behind the construction of the prototype. This phase was followed by a product evaluation stage where the team conducted various tests on the prototype to better understand the system and how it was able to function. After extensive testing and building two other robots, the team entered the final stage in the design cycle which consisted of the design finalization and summary. Here, the team implemented any additional features that were needed post-testing and reconstructed each robot to function as intended. This stage of the project involved a lot of self-reflection and revision of past ideas. Many of these thoughts are presented in the lessons learned subsections for each subsystem as a way to pay it forward to upcoming teams that plan to undertake this project.

2. Research, Benchmarking, and Competitive Assessment:

2.1. Resources Utilized for Research

During this project, a litany of resources was used to gather information and address the core challenges of aircraft maneuverability in hangars. These resources varied in type and focus, depending on the phase of the project and the specific technical challenges faced. Below is a summary of the key resources used for research:

2.1.1 Customer and Professional Insights:

The team's initial efforts were directed towards getting information directly from potential customers and other stakeholders. The first form of which were interviews with pilots at Mansfield Municipal Airport. The team took a trip there to interview four pilots to provide valuable information, such as accounts of aircraft movement processes, as well as any issues with current infrastructure and other useful information. These interviews informed the problem statement, highlighting the need for a system that prioritizes safety for both the plane and pilot, ease of use, and minimal physical strain.

In addition to this, the team also consulted numerous pilot and hangar forums, such as Pilots of America [2-1] and Beechtalk [2-2], where pilots shared their experiences and frustrations with current aircraft and manoeuvring methods. This helped the team further define the scope of the problem, increasing a particular emphasis on hangar rash (unseen damage from collisions) and the large turning radii of the aircraft being moved in and out of hangars.

The team also visited the Amazon Robotics facility in North Reading to draw inspiration from the robotic swarm systems in use there. The principal engineer who the team connected with, Gabe Hebert, was very helpful, giving the team ample time to look around and introducing several concepts for benchmarking, such as the pairwise comparison matrix. In addition to this, Amazon Robotics helped shape the mobility concepts for the robotic system, particularly drawing inspiration from their use of differential drive and fiducial vision.

2.1.2 Technical Papers and Other Resources:

The team also found some literature to aid in the project's development. These papers are listed below.

- *T. Ueyama, T. Fukuda, and F. Arai, \Conguration of communication structure for distributed intelligent robot system," in Proc. IEEE Int. Conf. on Robotics and Automation [2-7]*
- *G. Dudek et al., \A taxonomy for swarm robots," in IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, (Yokohama, Japan), July 1993. [2-8]*

- *M. Mataric, \Minimizing complexity in controlling a mobile robot population," in IEEE Int. Conf. on Robotics and Automation, (Nice, France), May 1992 [2-9]*
- *Woodman, Oliver J.. "An introduction to inertial navigation." (2007). [2-10]*
- *Diebel, James. "Representing Attitude : Euler Angles , Unit Quaternions , and Rotation Vectors." (2006). [2-11]*

These insights were pivotal for the D.U.N.G. project. The majority of these papers detail the underlying mechanics behind swarm robotics. This information is incredibly helpful, as it allows the DUNG project to stand on the shoulders of giants and use pre-existing infrastructure and methodology. Furthermore, the final two papers (Woodman and Diebel) provide a useful guide for navigation and representing rotations, which will be helpful when designing the controller for this project in the future.

In addition to the technical papers used to further the team's progress, pairwise decision matrices were used throughout the project to evaluate different design alternatives for mobility, sensing, and software, ensuring that all decisions were based on objective criteria such as cost, reliability, or power consumption. These matrices were shown to the team during the Amazon Robotics visit and so were quickly adopted into future workflows.

2.2 Existing Products or Processes Researched

The team conducted a comprehensive review of existing aircraft movement systems, including both motorized tugs and autonomous systems. Below is a summary of the key products and processes researched, with a focus on **best-in-class solutions** and **gaps in the market**.

Small Motorized Dollies and Tugs:

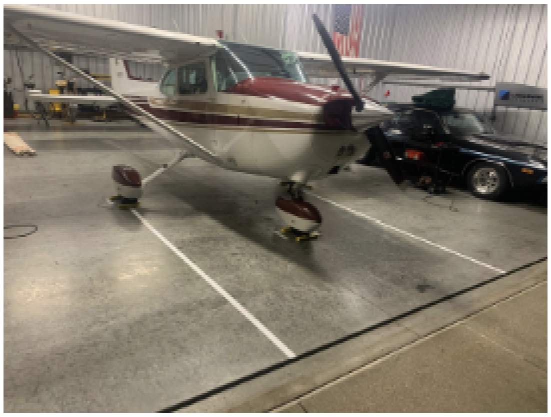


Figure 2.2.1: Image with Markers for Airplane Alignment

The main niche of devices that would compete with this project is motorized dollies or tugs. The user experience of these devices is as follows: The tug is secured to two pins on the side of the plane's front landing gear; the tug then actuates the aircraft by either rotating the front wheel or by rotating a wheel of its own and pulling the aircraft forward [2-1]. The aircraft can then be moved in or out of the hangar without the need for a great deal of physical force to push and pull the plane. However, it still requires thinking and planning to ensure that the aircraft avoids all obstacles.

This has led to hangars having some sort of visual indicator for alignment, such as tape laid across the floor, painted lines, or wheel endpoint markers This is shown in Figure 2.2.1.

In addition, these devices vary greatly in size, torque output, and power requirements. For example, one family of these devices is powered by power tool batteries, is small enough to be portable for a larger plane, and costs around $2000 [2-2]. Some of these devices employ a knurled rod to actuate the aircraft by turning its front wheel. On the other hand, another device simply attaches to the front wheel and tows the aircraft forward by spinning its own wheel [2-3]. Both of these types of device are also significantly less effective in the winter and rain. Both of these types of smaller devices also have issues with the lips of hangars, where pushing it up the small lip of the hangar is made significantly more accessible with the purchase of small rubber ramps that make this small change in height into a slight slope. Images of the knurled rod tug and multiple tire tug are shown below as Figure 2.2.2-2 and Figure 2.2.2-3, respectively.

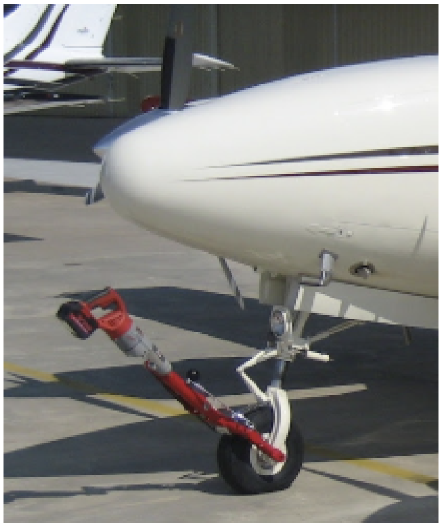


Figure 2.2.2: Plane Mover with Knurled Rod

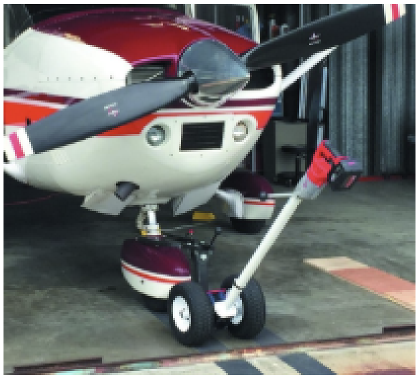


Figure 2.2.3: Plane Mover with Extra Wheels

Large Motorized Dollies and Tugs:

A larger family of these devices also exists that use the same actuation principles but with larger hardware. These devices are functionally the same but are powered by a diesel generator, mains power, or large batteries. The vast majority of these devices are not portable. An example of an aircraft tug in this category is an Aircraft Caddy 4K Junior, can move a 4,000lb aircraft and costs around $4000 [2-4]. An image of the Aircraft Caddy 4K is shown below as Figure 2.2.2.

Remote Control Tugs

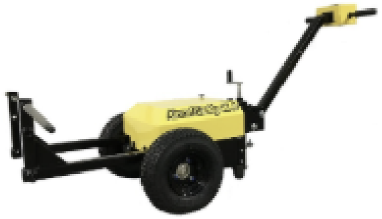


Figure 2.2.4: Aircraft Caddy 4K

Lastly, the team's research unearthed the existence of an emerging market of remote control tugs; for example, the AC TrackTech T1V2 - Compact Tug in the image below [2-5]. These devices have some internal intelligence, as they can prevent collisions between the tugs and obstacles, but this accident prevention feature begins and ends with tug collision. These tugs are also designed to carry loads of up to 21,000 lbs on the most expensive model ($12,000), and up to 2,000 lbs on the cheapest model ($5,000). These tugs are powered by a rechargeable lithium-ion battery and have a maximum speed of 1-1.5mph. These devices cradle the aircraft's wheels and drag it forward using two continuous tracks on either side. Depending on the aircraft's hold size, some of the smaller models are compact enough to be portable. An image of this type of tug is shown below in Figure 2.2.5

.

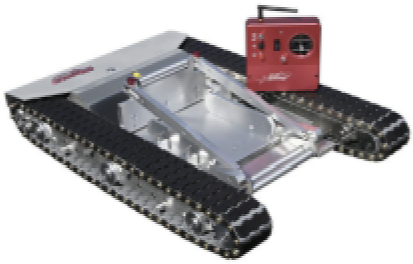


Figure 2.2.5: AC TrackTech T1V2 - Compact Tug

Amazon Robotics Mobile Robots

After the visit to Amazon Robotics, the team was inspired by Amazon Robotics' work. Amazon Robotics currently has two types of mobile robots: Hercules and Proteus. Hercules is an older model but is used throughout Amazon warehouse facilities, with over 250,000 units deployed. This robot is incredibly efficient, using fiducial vision to navigate and differential drive as its mobility solution, giving it holonomic motion and, thus, full freedom to move around. This robot is also able to carry a large pallet loaded with packages from one end of the warehouse to the other with no issue. The only problem with these robots is that they cannot be used around humans, as there is no way for them to detect a human, and this could lead to a significant accident.

Proteus is a robot designed to solve this problem by adding a litany of sensors and a human-robot interface that will alert a human if it is in the way or if an emergency is taking place. Proteus maintains almost all of Hercules' design aspects but is restricted in its movements to only move in the direction of where its interface is facing.

These two designs significantly inspired the team, as they are the most widely used mobile robots available and are physically capable of everything that the DUNG project hopes to accomplish.

2.3 Summary of Key Findings

Regarding competition, the existing tug and remote-control tug solutions adequately reduce users' physical mobility requirements. However, it still requires significant user forethought, and the user experience is made significantly smoother with the presence of another person to ensure that the pane is aligned and will not collide with another object.

Therefore, whilst several existing products have made the movement of aircraft in a hangar significantly easier, there are still some problems that remain unsolved by the competition. Firstly, all of these solutions require significant intervention by a human user to actuate the plane. In addition to this, none of the current solutions on the market can detect or predict collisions between the aircraft's body and the hangar, meaning that "hangar rash" (damage inflicted by the hangar on an aircraft) is still a significant issue for all of these solutions if used improperly or too hastily. Moreover, in the case of a community hangar, if a plane at the back of the hangar needs to be moved to the front, all of these solutions will take significant time and effort to use. Thus, there exists a significant gap in the market to be filled by this project.

2.4 List of Links and References Used

1. **Customer Insights**:
   - Mansfield Airport Interviews
   - *Pilots of America* forum: www.pilotsofamerica.com
   - *BeechTalk* forum: www.beechtalk.com
2. **Technical Reports and Competitor Products**:
   - *T. Ueyama, T. Fukuda, and F. Arai, \Conguration of communication structure for distributed intelligent robot system," in Proc. IEEE Int. Conf. on Robotics and Automation*
   - *G. Dudek et al., \A taxonomy for swarm robots," in IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, (Yokohama, Japan), July 1993.*
   - *M. Mataric, \Minimizing complexity in controlling a mobile robot population," in IEEE Int. Conf. on Robotics and Automation, (Nice, France), May 1992*
   - *Woodman, Oliver J.. "An introduction to inertial navigation." (2007).*
   - *Diebel, James. "Representing Attitude : Euler Angles , Unit Quaternions , and Rotation Vectors." (2006).*
   - *Aircraft Caddy 4K*: www.djproducts.com/product/aircraftcaddy-4k-jr/
   - *AC TrackTech T1V2*: www.acairtechnology.com

3. Customer Requirements

The most common priority of all pilots is making sure the aircraft does not collide with anything, as even a small collision can cause unseen damage which can manifest over time. While each interviewee had different ways of combating the issue, they all made sure that the aircraft was within a safety margin of some kind. In addition, almost all interviewees used a motorized device to make the actual moving of the plane less physically demanding. The relative importance of requirements would be: respect the safety margins of the aircraft, be maneuverable, and easy to set up, use and detach. The overall goal is to create a system which will minimize risk further, as well as eliminate any physical effort remaining.

All interviewees had multiple years of flying experience. Their personal aircraft hold anywhere from one flyer to at most 10. Each interviewee had their own personal hangar which they customized according to their own tastes. As a result, each hangar has different levels of clutter, other airplanes, and tools inside. Four individuals were interviewed. The interview transcripts can be found in Appendix 3.

The first person interviewed was Kenn D. Sebesta, who owns two airplanes and a glider in his hangar, which also contained a multitude of other items inside. His method of moving an aircraft around involves using a manual tow bar, which can be attached to the front landing gear. The airplane can subsequently be moved around by pulling the plane from the landing gear via the tow bar. Although he personally uses the manual method, he expressed that motorized options and fully standalone remote control models utilized by community hangars exist. His primary concern when asked about the biggest worry when it comes to moving airplanes is making sure the aircraft does not sustain unnecessary damage. The phenomena of an aircraft developing unforeseen damages after seemingly harmless collisions while on the ground is known as “hangar rash” and was the most common grievance listed by all interviewees. Kenn also expressed that time was not an immediate concern, since pilots using community hangars will usually phone in the night before the day they want to fly. Pilots do this so that the ground crew can bring their aircraft out for them with ample time and care, as this process is lengthy and time consuming. Kenn prioritizes the plane's safety and says that once an aircraft is moved outside, it should not obstruct the work of others.

The second person interviewed was Steve, an older pilot storing a single aircraft in a relatively clean hangar with little other obstructions. He explained that using a manual tow-bar was not the right fit due to the physical requirement of pulling a heavy aircraft on his own. He elects to use a motorized tow bar arm, the core of which is an electric power drill. After attaching the arm to the landing gear, a knurled wheel uses friction to turn the landing gear wheel, propelling the plane itself. Steve expressed that an airplane is difficult to move solo, and so the motor helps with reducing much of the physical strain. However, in rain and winter the knurled wheel cannot properly grip onto the plane due to the reduced wheel friction, and so Steve installed a winch on the rear of his hangar in order to tow the airplane back in. Additionally, the battery life of the arm is also a significant constraint, as Steve stated he could only activate the arm for four “movements”, or moving the plane in or out of the hangar. The arm can be easily disassembled and stored in the nose compartment, and he said he paid around $1800 for it. Steve overall prioritized using a movement system that is not physically strenuous, able to be easily deployed, and does not compromise the aircraft’s safety limits, mainly its landing gear turn limit.

The third person interviewed was Thomas, who owned a two-seater aircraft as well as a powered glider. Thomas had a comprehensive personal system in place to ensure accurate placement of his aircraft. Lines were laid on the ground with tape marking the trajectory of the landing gear, as well as where the glider and the two-seater aircraft should be located within the hangar. In addition, a tape diamond was placed on the far wall to be used as an alignment point while taxiing the aircraft back into the hangar. Thomas also uses a motorized pushing arm which attaches to the front landing gear of his aircraft. However, rather than turning the landing gear directly, it uses its own two wheels as a “tug” to push/pull the aircraft from. When he is away from his hangar, he uses a manual tow bar similar to Kenn’s to maneuver his aircraft. Thomas emphasized that his primary reason for the exhaustive hangar setup was to have ensured safety margins, as well as trying to eliminate situations where collisions can happen. He holds aircraft safety as the highest concern, as well as making sure to line up the aircraft’s pathing as much as possible, according to Thomas-it is harder to make precise adjustments the closer the aircraft gets going into the hangar.

The fourth person interviewed was John, who had a single aircraft, his tools, and three different kinds of motorized tugs. He expressed that the electric tug was the easiest to use since it was lighter. Along with Kenn and Thomas, the safety of the aircraft was his priority. He had relocated the plug which powers the hangar door close to the switches which operate the door itself, to make sure that he doesn’t accidentally close the door onto the aircraft. While not commenting as much as the previous interviewees, John expressed that the aircraft moving process was very involved, and that you should have three individuals around when moving an aircraft: one to move it and two to check the surroundings of the plane.

From the customer perspective, their requirements have been compiled into the table shown below, with each requirement having a minimum, and a desired satisfaction level.

Requirement	Acceptable	Desired
Airplane Safety	Able to avoid static obstacles and will stop if a discovered obstacle is detected.	Able to avoid static obstacles and respond to any discovered obstacles without stopping.
Maneuverability	Can maneuver an airplane without the need for a person's direct input	Can maneuver an airplane without the need for a person's direct input in any direction (semi/fully holonomic motion)
Easy to Use	Minimal input from the user, excluding initial setup.	Almost no input from the user, excluding initial setup.
Accuracy	Can position one aircraft in a single airplane hangar, or multiple airplanes a safe distance from each other.	Can position multiple airplanes in close proximity without contact.

Table 3.1: Customer Requirements with Thresholds

4. Conceptual Design Summary

4.1 Concept Generation Summary

For concept generatio, the primary issues to address are maneuverability, controllability, and ease of use. It discusses various solutions, including fully autonomous and semi-autonomous options, as well as subsystems like mobility, lifting, sensing, and software. The goal is to lay the groundwork for a robust final robot design that meets these operational needs.

4.1.1. Concept Overview

The team identified and segmented the project into major operations and components. This approach, along with a visit to Amazon Robotics for insights into their design process, helped guide brainstorming sessions and idea generation. Key design challenges include selecting between autonomous and semi-autonomous systems, and determining the number of robots required.

4.1.2. Mobility Concepts

Several mobility systems were explored to enhance D.U.N.G.'s maneuverability:

- Casters and Differential Drive: This method uses powered wheels and passive caster wheels. A challenge is ensuring all wheels stay grounded for stability
- Fully Motorized Casters: Four motorized casters allow omnidirectional movement but require precise synchronization to avoid mechanical strain
- Tricycle Skid Steering: Involves three wheels in a triangle configuration, with a steering caster at the front. Control issues may arise if the steering alignment is off
- Non-Wheel Methods: Alternatives like a flexible sling or air-based levitation were considered, each presenting unique challenges in stability and control.

4.1.3. Lifting and Attachment Concepts

Various methods for securing and lifting the aircraft were proposed:

- Sling: Flexible straps to lift the aircraft but risks instability and requires a strong fuselage
- Ramp/Cradle: The aircraft is placed into a cradle for lifting, but proper alignment is critical to prevent misplacement
- U-Bar: A locking mechanism around the landing gear for easy engagement, though challenges include ensuring even pressure and proper fit
- Scissor Lift: Provides a stable lift, though it requires significant clearance under the aircraft, and complexity increases with hydraulic or pneumatic systems

4.1.4. Sensing Concepts

To ensure safe navigation and collision avoidance, various sensors were considered:

- Encoders: Track wheel position but are susceptible to inaccuracies from wheel slip and terrain variation
- IMU: Measures acceleration and angular velocity, but gyroscope drift is a challenge. Kalman filtering can be used to maintain accuracy
- LiDAR: 2D and 3D LiDAR sensors were explored for obstacle detection. 2D LiDAR is cost-effective but limited in detection, while 3D LiDAR offers greater detail at a higher cost
- Computer Vision & Ultrasonic Sensors: Vision systems using fiducials offer precise positioning, while ultrasonic sensors are ideal for close-range detection in complex environments

4.1.5. Software and UX/UI

Software considerations depend on the system's autonomy level:

- Embedded Software: Both autonomous and semi-autonomous systems require software to manage hardware elements like motors and sensors. In a swarm model, communication protocols are critical for coordination
- Desktop & Mobile Apps: Both versions include a web and mobile interface for scheduling, maintenance tracking, and real-time updates. The mobile app offers flexibility for manual overrides in semi-autonomous setups

4.2. Downselection Summary

The concept selection process for this project, D.U.N.G., employed a "Divide and Conquer" approach, addressing major technical challenges: Mobility, Lifting and Attachment, Sensing, Embedded Systems, and UI/UX. Decisions were made using the pairwise comparison method, where each subsystem's concepts were scored based on predefined criteria. This approach enabled the team to evaluate options objectively and arrive at a solution that balances technical feasibility, cost, and user safety. The decision-making process was formalized through the creation of selection matrices, ensuring the input from all team members was incorporated.

4.2.1. Mobility

The team selected a system consisting of two motorized casters and two passive casters arranged in a staggered configuration. This design allows for holonomic motion, providing enhanced maneuverability. The motorized casters are capable of rotating 360 degrees, facilitating precise control over the robot's movement. Other alternatives, such as fixed drive wheels or a ceiling-mounted gantry, were considered but rejected due to higher costs, complexity, or limited functionality. The main challenge for the caster system lies in aligning the drive wheels to ensure straight movement. If misaligned, the robot may drift, affecting both pathing and safety. However, the system offers a robust solution with manageable technical complexities.

4.2.2 Lifting and Attachment

For the lifting and attachment mechanism, the ramp/cradle design was selected for its simplicity, reliability, and cost-effectiveness. The ramp facilitates easy alignment with the aircraft's landing gear, while the cradle secures the plane using a locking system. This passive mechanism minimizes the need for complex actuators, improving reliability and reducing maintenance costs. While the U-Bar design was considered, it was ultimately discarded due to its potential for misalignment and its lack of flexibility in handling various landing gear geometries. Extensive testing was conducted to ensure the ramp/cradle system can accommodate different aircraft sizes and perform with high precision.

4.2.3. Sensing

The sensing system was chosen based on criteria such as accuracy, cost, reliability, and integration complexity. After evaluating several options, the team decided on a suite of sensors, including an IMU (Inertial Measurement Unit), encoders, fiducial vision, and 2D LiDAR. These sensors were selected for their ability to support Simultaneous Localization and Mapping (SLAM), enabling the robot to navigate and detect obstacles efficiently. The IMU and encoders are crucial for localization, while the 2D LiDAR assists in dynamic obstacle detection. A few risks, such as sensor drift and integration challenges, were identified and were addressed during testing to ensure optimal performance. Ultimately, only a few of these sensors were implemented on the prototype but a real system would likely involve the usage of all of these devices.

4.2.4. Embedded Systems

The embedded systems for the autonomous robot were designed with full autonomy in mind, based on the team's evaluation of labor intensity, cost, and safety. The decision to implement a fully autonomous system was supported by a pairwise comparison matrix, which highlighted the benefits in terms of efficiency and safety. The network topology chosen for communication between robots is a Star configuration, providing robust and easy-to-manage communication with a central hub. The A* path planning algorithm was initially selected for its simplicity and efficiency but the team ended up working with a Floyd-Warshall algorithm because it is more applicable to the group’s use-case.

4.2.5. UI/UX

For the user interface, the team opted for a web-based application due to its high scores in usability, reliability, and cost-effectiveness. The application features a secure login system, allowing aircraft owners and airport operators to schedule aircraft movements and maintenance. The user interface is intuitive, with minimal training required for operation. The web app's advantages include easy accessibility and integration with the robot's embedded systems. While desktop and mobile applications were also considered, the web app was deemed the most suitable for the project's needs, offering flexibility and ease of deployment.

4.3. Evaluation of Conceptual Design

The conceptual design of the autonomous aircraft-moving robot had been substantially defined, with key decisions and system architectures established across all major subsystems. Mechanically, the team had converged on a semi-motorized caster wheel arrangement for mobility, incorporating two motorized caster wheels and two passive casters arranged in an “X” formation. This configuration offered the maneuverability needed to navigate complex hangar environments while maintaining a manageable level of complexity and cost. For lifting and attachment, the ramp/cradle system had been selected over alternatives like U-bars or scissor lifts. Its stable engagement, broad compatibility, and relative design simplicityl helped ensure both user and aircraft safety while minimizing issues related to durability, performance, and maintenance.

On the sensing front, the suite of sensors had been carefully refined to balance accuracy, reliability, and affordability. The integrated approach included an IMU for orientation and movement data, encoders for precise positional tracking, and fiducial vision markers to reinforce localization capabilities within the hangar. This was planned to be complemented by 2D LiDAR and triangulation methods for obstacle detection and mapping. Together, these sensing solutions would theoretically have been able to support SLAM (Simultaneous Localization and Mapping), allowing the system to effectively identify and avoid dynamic obstacles while maintaining high precision during aircraft positioning tasks. During later stages of the project, the sensing suite was downscoped to account for budget constraints.

Progress in the embedded systems domain at this stage had been equally significant. The decision to operate the robots fully autonomously set a clear direction for control logic and software integration. The star network topology had been selected to facilitate robust and reliable communication throughout the robotic swarm. In tandem, the A* path-planning algorithm had been chosen for its simplicity, reliability, and ease of implementation. Although A* is not inherently dynamic, proper environmental modeling and scenario planning will help mitigate this limitation, ensuring the robot can efficiently navigate within a complex hangar setting. As discussed in the appropriate section, this decision was revised after speaking with representatives at Amazon Robotics who advised the team to switch to Floyd-Warshall to simplify integration with other systems.

From a user-interface perspective, the conceptual decision favored a web-based application. This platform provided a practical balance of accessibility and maintainability, with secure login capabilities, visual hangar layouts, scheduling options, and control features that ensure aircraft movements are intuitive and easy to manage. Although desktop and mobile applications remained close contenders, the web-based approach promised seamless integration with both back-end databases and the on-robot systems, as well as simpler update cycles and broader device compatibility.

With the conceptual design largely completed, substantial tasks remained focused on refining and implementing the chosen concepts. CAD models were finalized to ensure that dimensions, tolerances, and material selections meet performance requirements.

5. Product Generation

5.1. Material Selection

5.1.1 Chassis Materials

The first material choice was aluminum extrusion for the frame of the system. There were multiple reasons for this decision, the first of which was accessibility. Aluminum extrusion, especially on the length scale of this prototype is readily available at workshops at Boston University. Furthermore, extrusions allowed for a versatile mounting system that was adaptable to the inevitable improvements that were made to the prototype system. Aluminum as a material is sufficiently rigid on the scale of forces that this prototype can be expected to experience, especially given its yield strength of 241.1 N/mm^2 [5-1]. The Extrusions' straightness values of 0.32mm per 304mm of length and flatness of 0.1mm per 25.4mm of width [5-1] meant that they could be used without worrying about mateability or uniformity issues.

An idea that was initially considered was to machine a one-inch-thick plate of aluminum using a CNC machine to add the necessary mounting points for the motorized/non-motorized casters and the cradle mechanism. However, this is suboptimal for a prototype because machining an aluminum plate takes more time, is more expensive, and is a concrete design that can not be greatly adjusted. This idea does have the potential to be the form factor for the full-scale system. Further analysis would be needed, including a Finite Element Analysis of the full-scale system.

Another material selection was to 3D print the sub-mechanisms of the system with PLA. This was an optimal choice because it allowed for rapid prototyping of each component. At the scale of forces that the robot prototype is expected to experience, a 3D-printed PLA part will be sufficiently strong enough due to its "bending strength" as defined by Bambu Studios to be 76MPa [5-2]. The scale plane load was under 50 pounds and be distributed among 3 robots each with 4 wheels, far less than what is required for the permanent deformation of PLA.

5.1.2. Electronic Materials

The first electronic selection was the control board. The team selected the ESP-32 S3 for low-level control because of its ability to manage a sufficiently large number of general-purpose input/out pins, high processing power, a high-resolution analog-to-digital converter, and built-in low-latency communication protocols, such as ESP-NOW [5-3]. Next, a board that could interpret the camera for the fiducial vision was selected. This was the Raspberry Pi Model 4B. This board was chosen for its built-in camera functionality, its wireless communication ability, and its relatively low cost. In addition to these, an L298N motor driver board has been selected, as it is able to control two motors at variable speeds and is kept in stock by BU's RASTIC facility.

The remainder of the electronic materials are DC motors with built-in encoders for the driving of the wheels, as they have high torque, are relatively inexpensive, and have built-in sensors that can be used for real-time robot positioning. Two MG99R servo motors have also been selected for the steering of the robot, for similar reasons to the DC motors, as they are also relatively inexpensive, and require very little control infrastructure, as a simple ‘Move to a set angle’ command will suffice. The same can be said for the MG90S micro servo, which will be responsible for the locking of the cradle mechanism.

5.2 Process Selection

Another key consideration in process selection was the team’s Design for Manufacture (DFM) approach, ensuring that the prototype could be built using the facilities available at Boston University. By leveraging accessible manufacturing methods, the team streamlined the fabrication process while maintaining design flexibility. The chosen processes were 3D printing, simple drilling, cutting, and milling. These were selected based on the team members’ prior experience, allowing for efficient and precise construction without the need for outsourcing or specialized equipment. This approach ensured that the prototype could be iterated quickly while keeping production costs manageable.

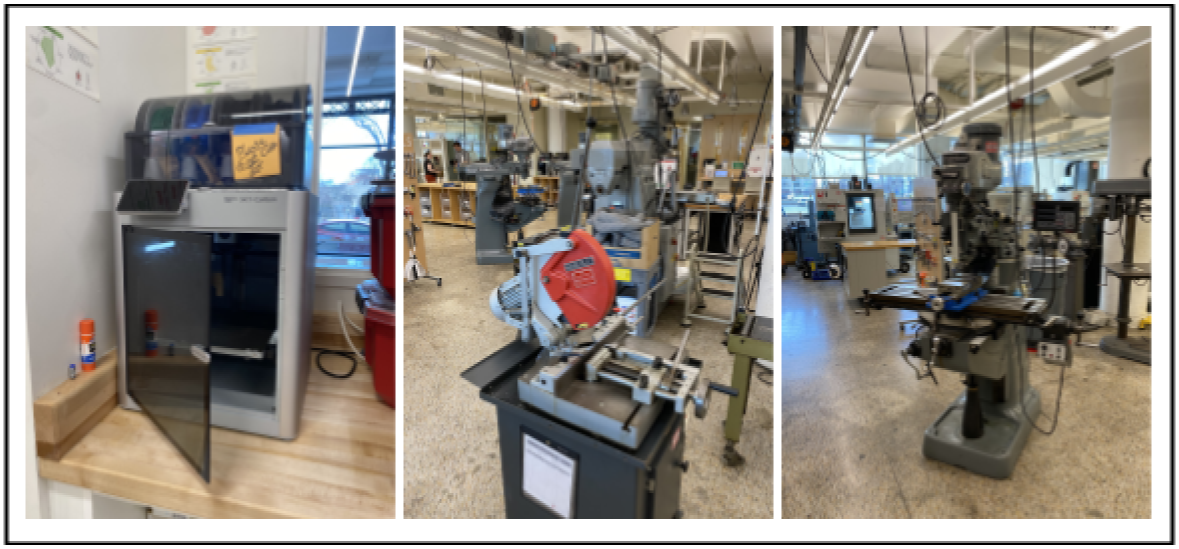


Figure 5.1: Equipment Used (Left to Right): Bambu X-1 Carbon, Chop-Saw, Milling Machine

For instance, 3D printing using a Bambu X-1 Carbon was extensively utilized for fabricating sub-mechanisms due to its rapid turnaround time and ease of modification. By using the Bambu, the team could quickly test and refine parts without the time and expense associated with more traditional machining methods. Additionally, straightforward machining principles were employed for the aluminum extrusions, where structural rigidity was required. These processes allowed the team to create precise mounting points while minimizing complexity, ensuring that the system could be assembled efficiently using readily available tools. This manufacturing approach facilitated the construction of the prototype and provided a basis for potential optimizations for future iterations of the system. The Chop-Saw was used to make all cuts in the aluminum extrusion because the end-stop that was built into the tool allowed for repetitive cuts to be made without variation. The mills were used to drill holes into the aluminum because the translational increment of the tool head was 0.003mm [5-4] which allowed the team to machine accurately and precisely.

Programming the software processes of the project involved free-to-use software which members had experience with such as MATLAB, JavaScript, C++, and Python. This decision was made to allow more time to be spent actively contributing to the project's goals rather than learning new approaches. The major development environments used include Nano for Python, the Arduino IDE for C++, and the MERN Stack approach for JavaScript.

5.3. System Design Approach

5.3.1. Process Flow

Before setting out to build the software aspects of the project, the team constructed a flowchart describing how the system is meant to operate and how various components communicate. This approach allowed team members to discuss their ideas and reach a common conclusion. Additionally, this chart gave insight into what each software section will communicate and what format subsystems should expect to receive and transmit data in.

A linear processing system was selected for its simplicity. Though this approach is error-prone and limiting due to the lack of feedback, it allowed team members to focus on the functionality of individual subsystems.

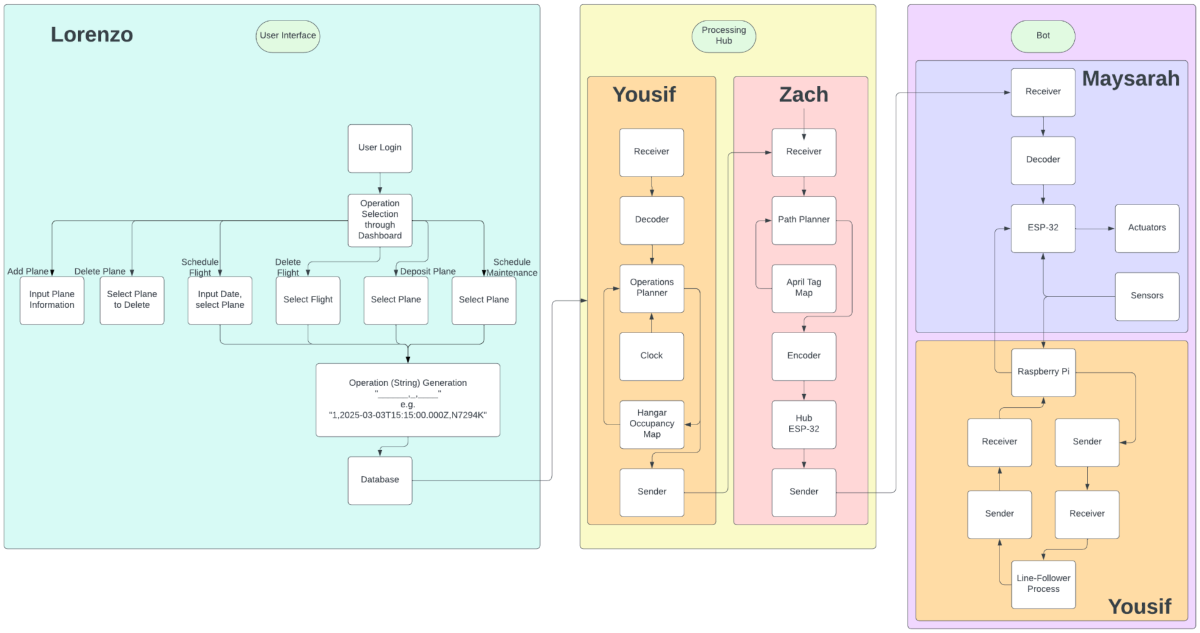
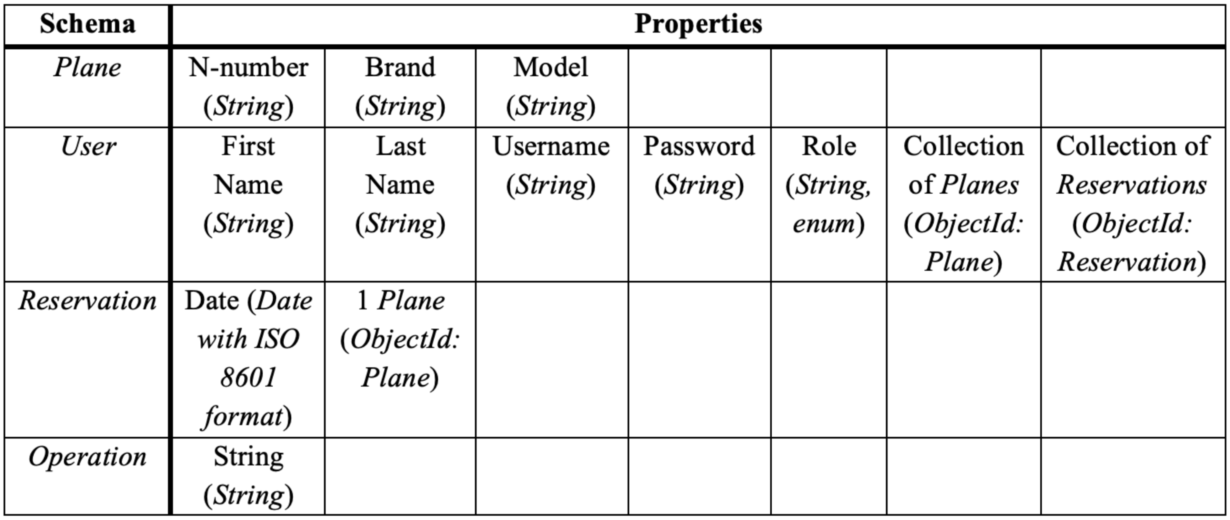


Figure 5.3.1: Process Flowchart (See Appendix 5.3.1 for a Higher Resolution Image)

5.3.2. Web-App Architecture and Considerations

To allow owners of planes and airport maintenance workers to easily access the DUNG system, a complete full-stack web application has been created by using the MERN stack (MongoDB, Express.js, React, and Node.js). The general architecture is based on a client-server model, with React used for the front-end framework (in simple terms, what is graphically displayed on the web app), Express.js and Node.js to power the back-end server (the logic behind the code) and API (the protocol to allow front and back-end to "communicate"), and MongoDB that works as NoSQL database for the storage. The team chose to use a MERN stack because it uses JavaScript for the entire ecosystem, is open-source, and presents good cloud compatibility. Moreover, it offers scalability, flexibility, and speed making it ideal for building modern, dynamic web applications. The software is not hosted on a remote server, but it runs locally on a computer. At this stage, DUNG does not store large amounts of data in the database, so the basic (free) version of MongoDB is sufficient, keeping the total software cost at $0.

To easily access and manage data, a Mongoose schema-based model solution has been implemented. All data is saved on the MongoDB database. The schema is used to represent planes, users, flight reservations, and operations. Table 5.3.1 summarizes all characteristics of each model.



Schema	**Properties**						
Plane	N-number (*String*)	Brand (*String*)	Model (*String*)				
User	First Name (*String*)	Last Name (*String*)	Username (*String*)	Password (*String*)	Role (*String, enum*)	Collection of *Planes* (*ObjectId: Plane*)	Collection of *Reservations* (*ObjectId: Reservation*)
Reservation	Date (*Date with ISO 8601 format*)	1 *Plane* (*ObjectId: Plane*)					
Operation	String (*String*)						

5.3.3 User Interface and User Experience (UI/UX)

Figure 5.3.2a shows the first graphic interface that allows users to register a new account by typing first and last name, username, and password; Figure 5.3.2c shows the checkbox with a secret code (only airport workers would have this) needed to allow the creation of a maintainer account. Figure 5.3.2b shows the login interface.

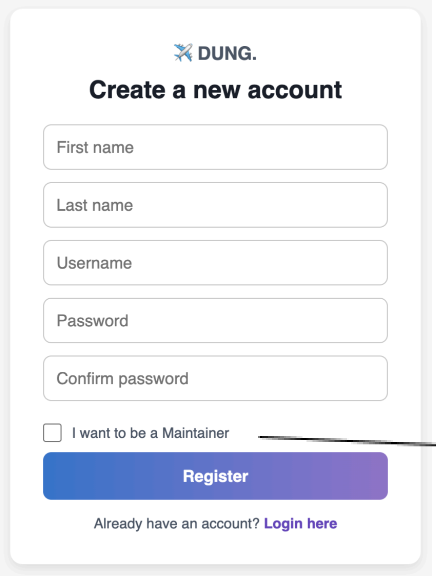


Figure 5.3.2a: Account Registration Form

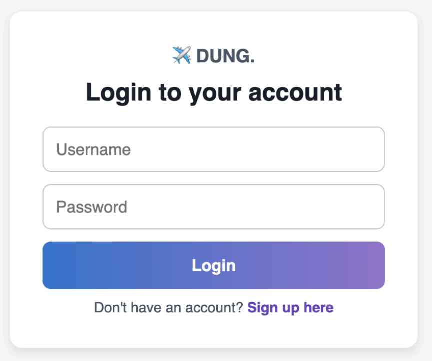


Figure 5.3.2b: Login Interface

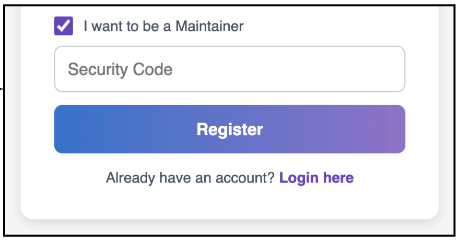

☑ I want to be a Maintainer
Security Code
Register
Already have an account? Login here

Figure 5.3.2c: Maintainer Registration Security Code

Once the owner of one or multiple planes is logged in (through tokenization the system "knows" who is logged), a dashboard appears (Figure 5.3.3), which shows all the main functionalities of DUNG.

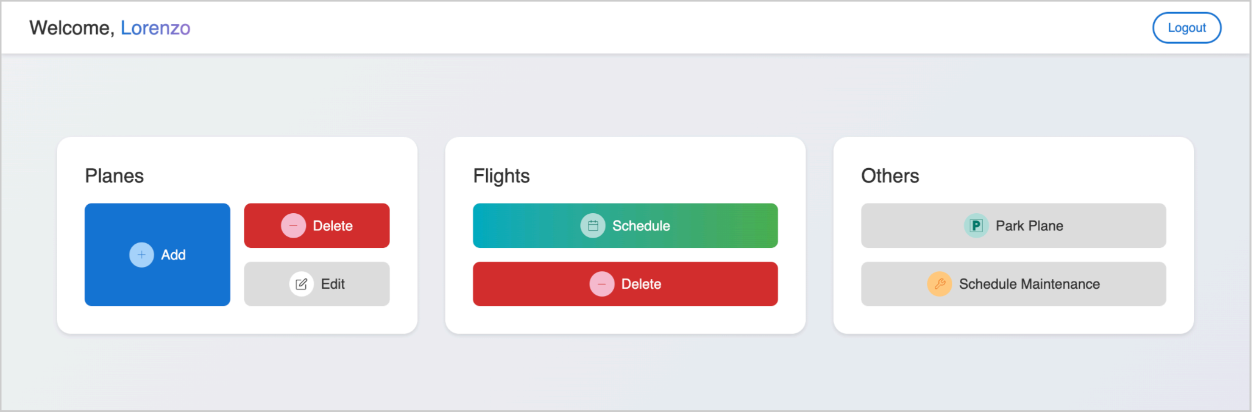
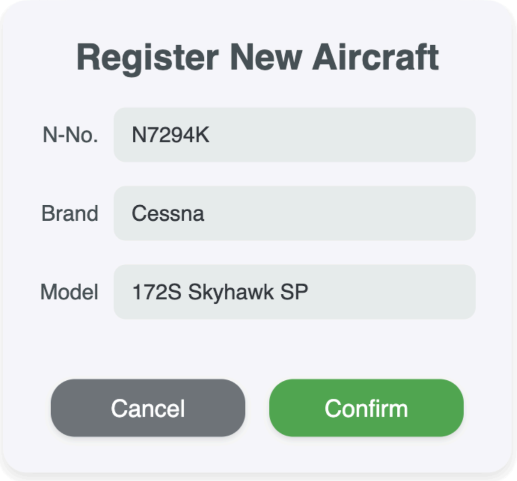


Figure 5.3.3: Standard User Dashboard

In the “Planes” section it is possible to add (Figure 5.3.4) or remove (Figure 5.3.5) a plane to the hangar, or edit an existing one (in development).



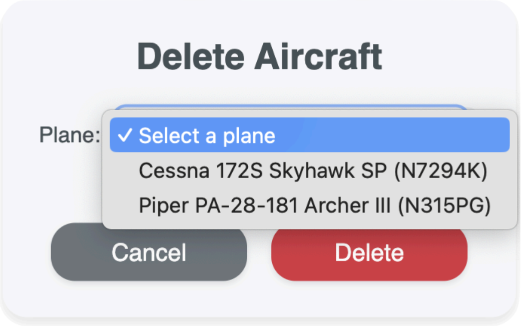
Register New Aircraft

N-No. N7294K

Brand Cessna

Model 172S Skyhawk SP

Figure 5.3.4: New Aircraft Registration



Delete Aircraft

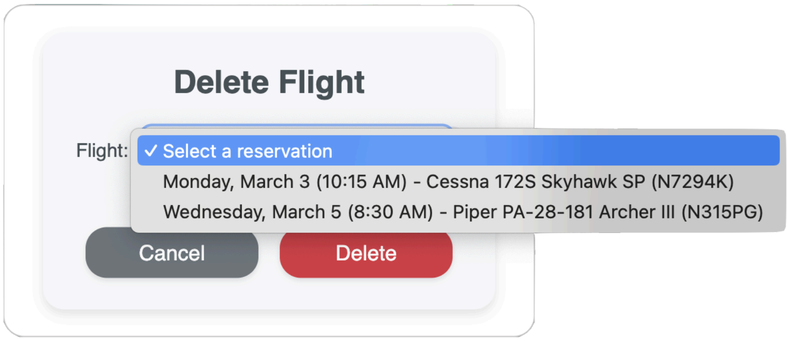
Plane: ✓ Select a plane

Cessna 172S Skyhawk SP (N7294K)

Piper PA-28-181 Archer III (N315PG)

Figure 5.3.5: Aircraft Cancellation

In “Flights” the user can schedule a new flight (Figure 5.3.7) or delete an existing one (Figure 5.3.6).



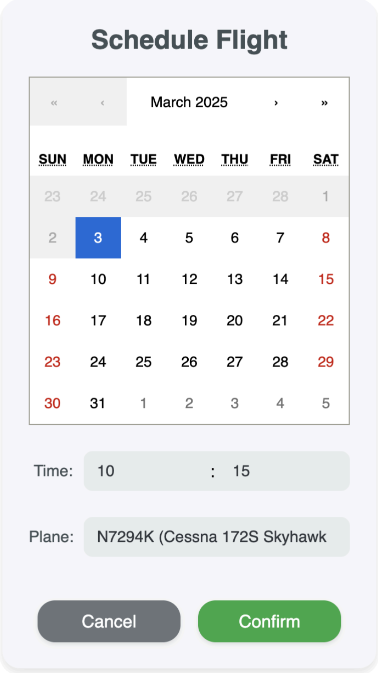
Delete Flight

Flight: ✓ Select a reservation

Monday, March 3 (10:15 AM) - Cessna 172S Skyhawk SP (N7294K)

Wednesday, March 5 (8:30 AM) - Piper PA-28-181 Archer III (N315PG)

Figure 5.3.6: Flight Cancellation



Schedule Flight

« ‹ March 2025 › »

SUN	MON	TUE	WED	THU	FRI	SAT
23	24	25	26	27	28	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

Time: 10 : 15

Plane: N7294K (Cessna 172S Skyhawk

Figure 5.3.7: New Flight Reservation

The *Park Plane* or *Schedule Maintenance* buttons allow the user to deposit a plane in a selected parking outside of the hangar and to plan maintenance for a specific plane, respectively.

Other functionalities are only available to airport workers, whose dashboard will be slightly different (Figure 5.3.8), with the hangar settings button that allows to set the maximum number of planes for the hangar, and an emergency button to shut down the entire system in case of a malfunction.

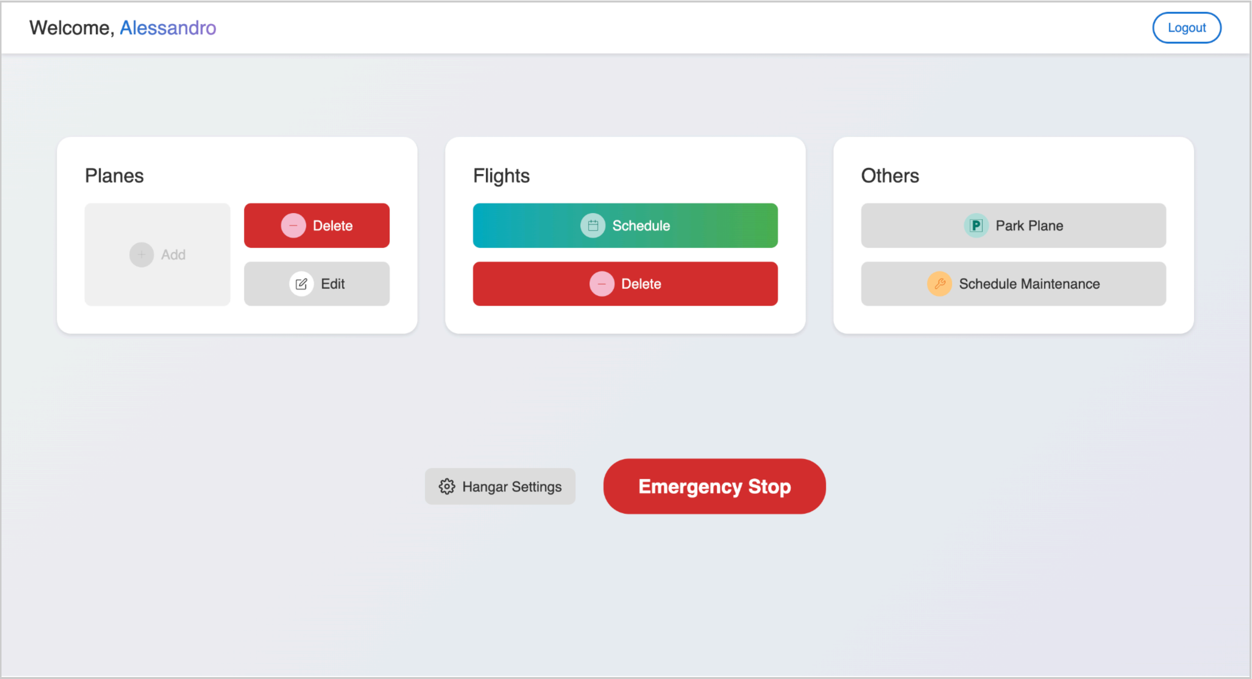


Figure 5.3.8: Maintenance User Dashboard

Another difference is that maintainers cannot add new planes (when the owner buys new planes) but can access all planes and flight reservations of all the owners, this means that the airport worker can delete any scheduled flight, delete any aircraft, or schedule a flight for any pilot (while a "standard" user can only manage their own planes and flights).

5.4 Motion Planning Considerations

In order to simplify the control environment, it has been determined that the simplest implementation that satisfies the necessary requirements is using a network of April tags located throughout the hangar. Some examples of critical April tag locations are at the base of the aircraft's landing gear, the charging stations, and checkpoints within the rotation area. Shown in Figure 5.4.1 is an older draft of the April tag network iteration which would have been used to map out the prototype hangar.

To find the optimal path in the April tag map, the Floyd Warshall algorithm [5-5][5-6] was used to find the shortest path. The input to the algorithm is a matrix that shows the "weights" or cost of traveling a path and the connections between points. A higher number corresponds to a higher weight which is less preferred. The

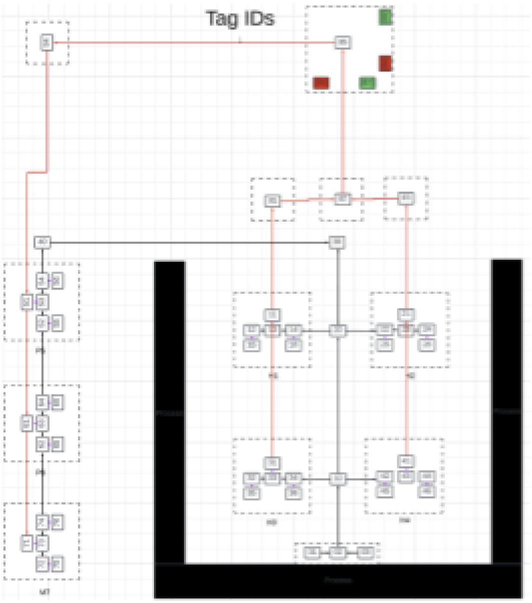
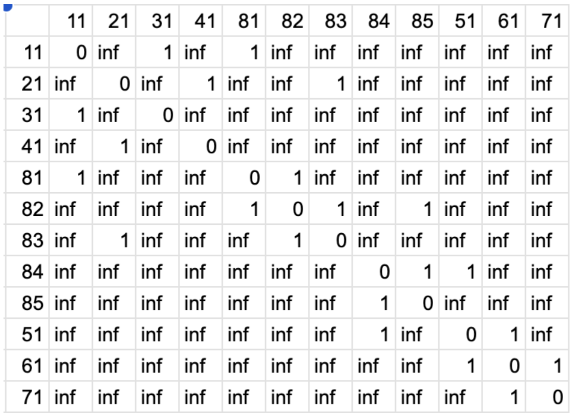


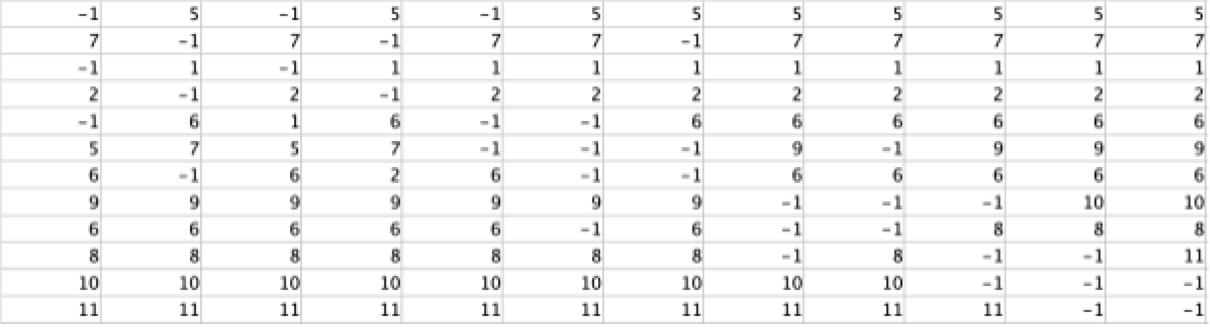
Figure 5.4.1: April Tag Map

algorithm will tally the total weighted sum and determine which one is the smallest for going from one point to another. It then outputs a matrix that will be called the "routing matrix". This matrix gives the index of the best "in-between" node that should be used for a path from point M to point N (for an MxN matrix). As a result, by continuously finding "in-between" nodes until the routing matrix returns a value of -1, indicating no more points in between are possible. Shown below is the routing matrix used for navigating the plane system. See Figures 5.4.2 and Figure 5.4.3 for the routing matrix and relationship matrix.



	11	21	31	41	81	82	83	84	85	51	61	71
11	0	inf	1	inf	1	inf	inf	inf	inf	inf	inf	inf
21	inf	0	inf	1	inf	inf	1	inf	inf	inf	inf	inf
31	1	inf	0	inf	inf	inf	inf	inf	inf	inf	inf	inf
41	inf	1	inf	0	inf	inf	inf	inf	inf	inf	inf	inf
81	1	inf	inf	inf	0	1	inf	inf	inf	inf	inf	inf
82	inf	inf	inf	inf	1	0	1	inf	1	inf	inf	inf
83	inf	1	inf	inf	inf	1	0	inf	inf	inf	inf	inf
84	inf	inf	inf	inf	inf	inf	inf	0	1	1	inf	inf
85	inf	inf	inf	inf	inf	inf	inf	1	0	inf	inf	inf
51	inf	inf	inf	inf	inf	inf	inf	1	inf	0	1	inf
61	inf	inf	inf	inf	inf	inf	inf	inf	inf	1	0	1
71	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf	1	0

Figure 5.4.2: Relationship Matrix



-1	5	-1	5	-1	5	5	5	5	5	5	5
7	-1	7	-1	7	7	-1	7	7	7	7	7
-1	1	-1	1	1	1	1	1	1	1	1	1
2	-1	2	-1	2	2	2	2	2	2	2	2
-1	6	1	6	-1	-1	6	6	6	6	6	6
5	7	5	7	-1	-1	-1	9	-1	9	9	9
6	-1	6	2	6	-1	-1	6	6	6	6	6
9	9	9	9	9	9	9	-1	-1	-1	10	10
6	6	6	6	6	-1	6	-1	-1	8	8	8
8	8	8	8	8	8	8	-1	8	-1	-1	11
10	10	10	10	10	10	10	10	10	-1	-1	-1
11	11	11	11	11	11	11	11	11	11	-1	-1

Figure 5.4.3: Routing Matrix

5.5. Embedded System Configuration

In this project, the boards controlling the embedded system are an ESP-32 S3 and a Raspberry Pi Model 4B. The ESP-32 controls two servo motors and two DC motors and obtains readings from an MPU6050 IMU, the motor encoders, and the potentiometer for the cradle mechanism. The ESP-32 is responsible for syncing movements across different robots via the ESP-NOW communication protocol, which is low latency and works over both Wi-Fi and Bluetooth.

The Raspberry Pi is responsible for sending out a live video stream of the tape and fiducials that the navigation system relies on, as well as for communicating between an outside computer and the robot.

Therefore, to move the robot while in its current state, a computer must connect to the Raspberry Pi over wifi and send it commands. From here, a Python script interprets the user inputs and sends a string input to the ESP-32. The ESP-32 is loaded with a C++ program that interprets this input and moves the motors in the four cardinal directions, depending on the user's input. This is simply a testing program to understand the robot's motion and tune its parameters.

6. Product Evaluation

6.1 Embedded and Control Systems

Once the physical body of the first robot was built, members of the team worked on implementing the "mind" behind the operations of the robot. When the line following capabilities of the robot were initially tested using an idealistic model, it was found that the system would veer off-course often and lose its track. Additionally, the team realized that the robot would tend to overcompensate its trajectory leading to high overshoot and an unstable system. To address this issue, the team revisited the model and adjusted the parameters to more accurately represent reality. Additionally, a collective effort was made to observe the motions and calibrate a Proportional-Derivative controller for effective line-following. Though this approach is prone to steady-state error, it was decided that the addition of an integral term would require more work than it's worth given the urgency of getting a functioning prototype.

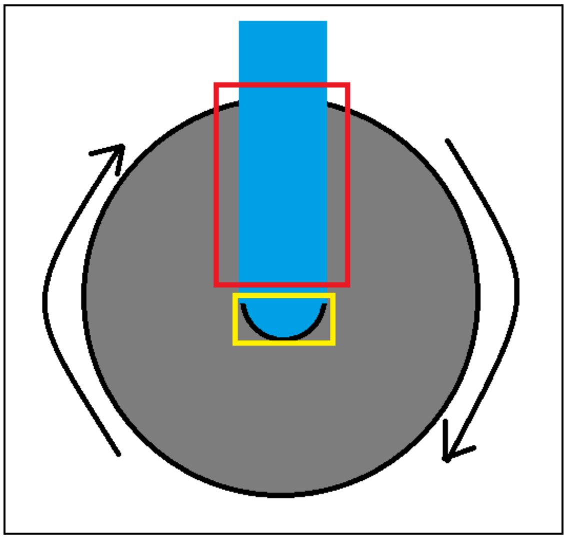


Figure 6.1.1: Points of Friction on the Wheel Assembly

Furthermore, while dealing with the robot's trouble with line-following, it was realized that the friction imbalances between the wheels led to issues with the controller-system interface. Figure 6.1.1 shows two regions where wheel-frame friction was prevalent. The red region shows where the supporting forks made contact with the surface of the wheels leading to stick-slip friction [6-1] and the yellow region describes where frictional forces were present due to overtightening the wheel fastener. This additional friction resulted in different wheels yielding differing results when given the same command, which meant that an entirely different control system had to be implemented to ensure the rate of rotation of the wheels matched the command that they were sent. To combat this problem, a full-fledged Proportional-Integral-Derivative [6-2] controller was used to achieve consistent outputs across all wheels. The addition of this controller also meant that tuning the line-following controller was a much simpler task, as drag forces between parts

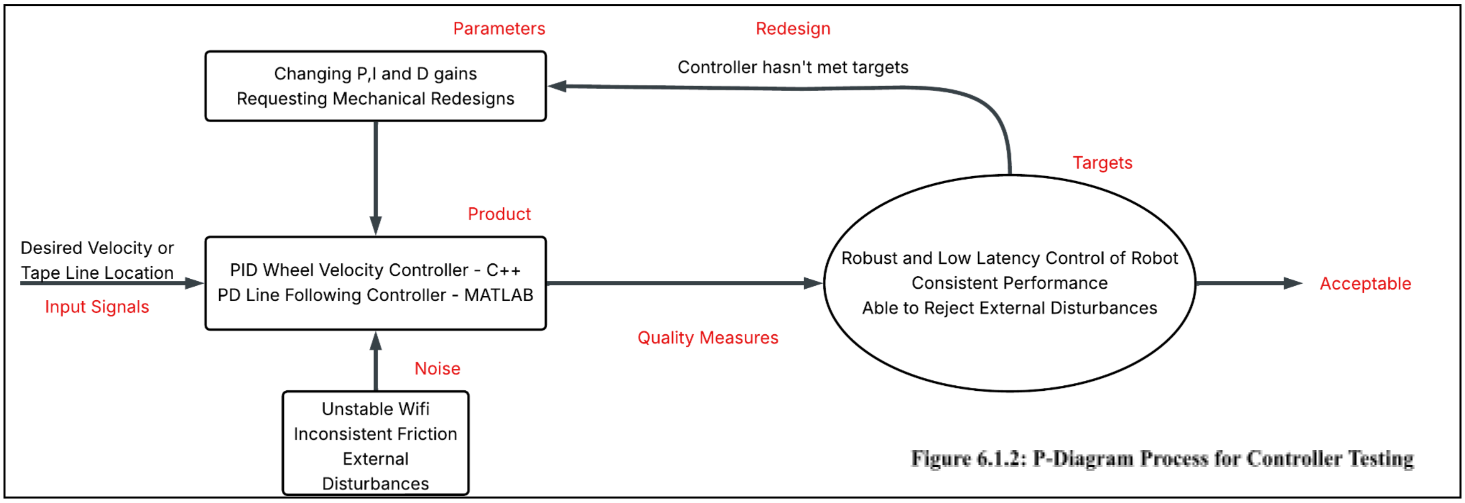


Figure 6.1.2: P-Diagram Process for Controller Testing

could be disregarded with minimal impact. Figure 6.1.2 depicts a P-diagram showing the process of tuning the PID controllers and how the targets were met.

Upon completion of the control system for the motions of the robots, more extensive testing was done to gauge how well the system could accomplish various tasks. Fortunately, the built-in controllers within the servo motors were able to consistently reach their desired target angles [6-3]. Additionally, with the development of a reliable April tag recognition system came its own set of challenges. During static testing, the program was able to highlight April tags with incredible accuracy. However, when the robot was instructed to follow a line and stop at a specified tag, it would often not recognize the April tag as it passed over it.

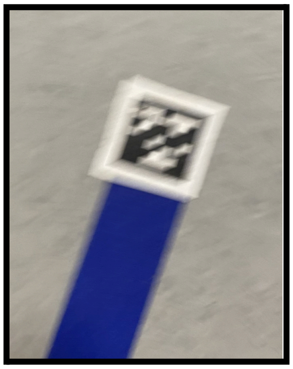


Figure 6.1.3: Motion Blurred April Tag

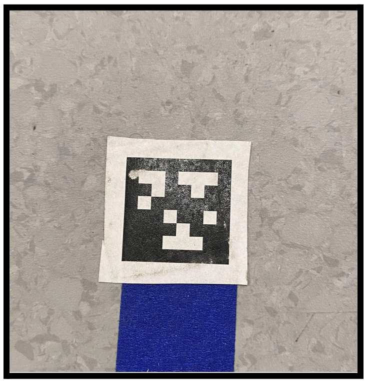


Figure 6.1.4: Snapshot of a Recognizable April Tag After System Adjustment

The issue resided in the low field-of-view of the camera that the system used and how fast the robot needed to move to avoid static-friction-related issues. There was a noticeable motion blur, as shown in the snapshot displayed in Figure 6.1.3, that rendered the recognition system blind to most fine details, leading it to miss its markers. The solution to this problem came in multiple forms. Firstly, a ring light was added around the lens to increase the visibility of objects in the field of view. Additionally, the source programming of the camera module was altered to increase its shutter speed and achieve sharper snapshots [6-4]. The result of these changes is shown in Figure 6.1.4. The mounting of the camera was also tilted to allow for vision ahead of the robot's position to facilitate feedforward control, as shown below in Figure 6.1.5.

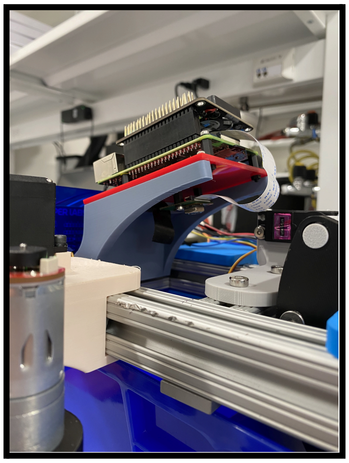


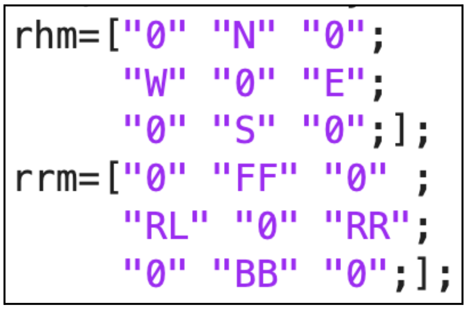
Figure 6.1.5: Upgraded Camera Mount

6.2. April Tags and Motion Planning

The motion planning software underwent major streamlining due to hardware revisions and changes made to the demonstration hangar. Software changes were made to account for robot rotation, the smaller mock hanger used in the demonstration layout, and the start of constructing the floorspace the robot ran on.

After extensive discussion and testing alongside mechanical and embedded systems subteams, the conclusion that the robot would have a difficult time performing sideways strafes to high precision was reached. Therefore, the robot instead rotated the plane to avoid strafing motions.

The motion planning software was then subsequently changed to substitute sideways strafing with in-place rotations. A virtual compass plots the heading of the robot as it travels through the hanger. As the robot only rotates at right angles, the virtual compass is represented as a 3x3 matrix, as shown in Figure 6.2.1. This matrix is rotated 90 degrees when necessary to align itself with the intended orientation of the system. Lastly, the orientation of this matrix is used to suitably rotate the robot [2-11].



```
rhm=["0" "N" "0";
     "W" "0" "E";
     "0" "S" "0";];
rrm=["0" "FF" "0" ;
     "RL" "0" "RR";
     "0" "BB" "0";];
```

Figure 6.2.1: New Rotation and Gyro Matrices

Along with software changes, the hangar layout was simplified for the demonstration. The original layout of the demonstration map consisted of four hangar bays, two parking bays, and a maintenance bay. The roadmap has been changed to incorporate only one of each critical element: one hangar bay, one maintenance bay, and one parking bay, all connected in a “T” configuration. The new configuration of the demonstration layout is shown in Figure 6.2.2.

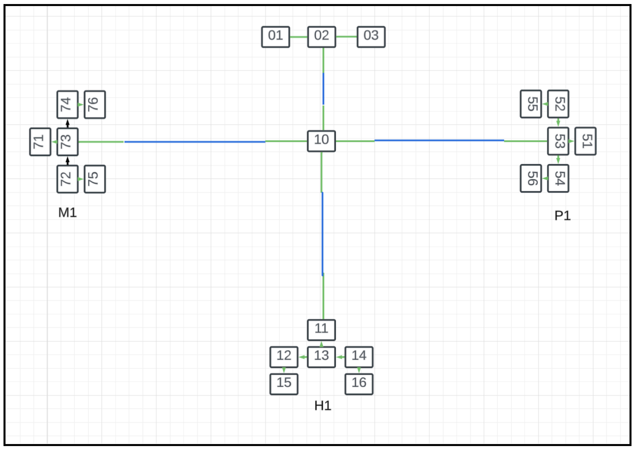


Figure 6.2.2: Revised Demonstration Layout with Correct Colors

The last changes made to the map are accommodations for the April tag recognition system. As previously discussed, the robot had trouble reading April tags while traveling. Therefore, slow zones have been incorporated into the motion planning in order to give the robot a clearer image when driving over an April tag. This change manifested itself in the form of green tape lines that precede each fiducial marker. This contrasted well with the existing blue tape, which was used for general-purpose pathing and combined with the above measures taken to improve the camera image quality, resulting in consistent readings of April tags. These new slow zones are represented visually in Figure 6.2.2.

6.3. Mechanical Design

The project's mechanical development was driven by an iterative cycle of design, testing, and refinement. Each prototype version revealed new challenges, enabled experimentation with different solutions, and led to improvements in performance and reliability. The initial goal was to achieve basic functionality. Once that was established, the focus shifted to strengthening the design, increasing consistency, and improving integration with the final system.

The frame underwent several early revisions, as shown in Figure 6.3.1. The team began with components screwed directly together, which functioned but lacked structural integrity. The project then transitioned to using channel corner brackets for improved support and later reinforced those with flat corner brackets that included underlying supports. This final version provided the necessary strength, particularly during movement and loading scenarios.

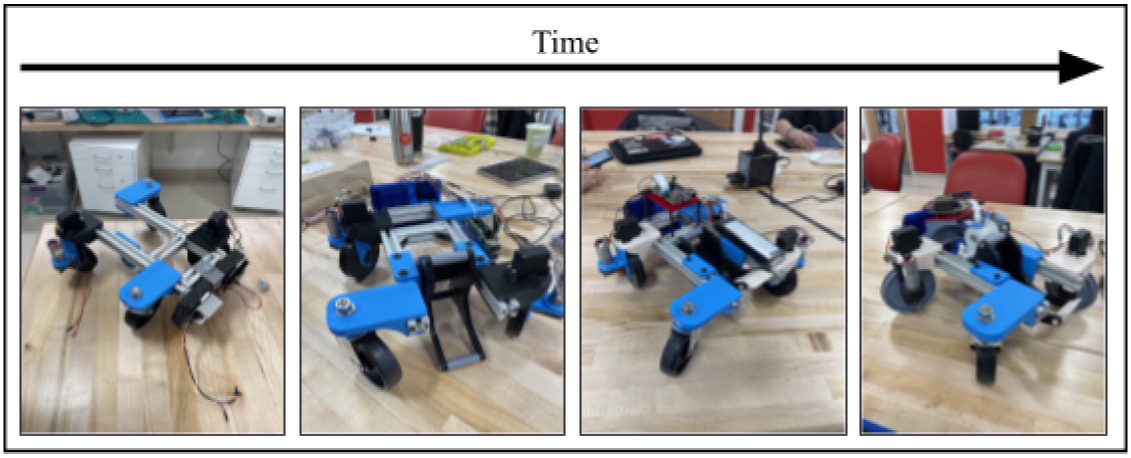


Figure 6.3.1: Mechanical Evolution of D.U.N.G.

The motorized caster posed more issues than anticipated. Initially, the wheel made contact with the bracket due to an O-ring, which was resolved by trimming nonessential parts of the bracket to create additional clearance. A more significant problem was lateral movement or "wiggle." The servo had originally been press-fit into a fixed bracket, but this allowed unwanted motion. To address this issue, the team removed the bearing and redesigned the rotating bracket to extend into the bearing hole. The team also replaced the press fit with a metal servo horn attachment, greatly improving stability. Tolerance issues with the bearings also emerged—some fits were too tight, others too loose. Crush ribs were added to the internal bearing mounts to help manage this variation, though results were still imperfect. The external bearing fit was resolved with a standard press-fit approach [6-5]. In addition, the caster wheel experienced excessive friction due to its wide contact area. Narrowing the wheel helped reduce rolling resistance and improved performance.

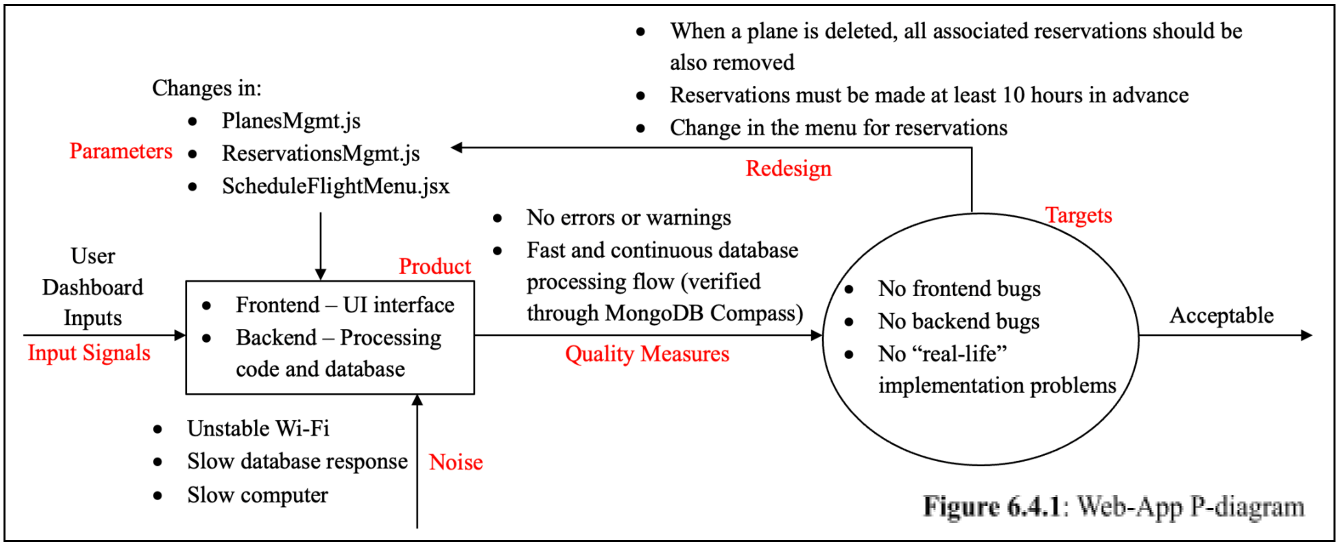
Substantial changes were also made to the cradle. Originally, it required too much force to insert an object. To address this, a two-sided ramp was added to help guide items in. While that improved entry, the cradle was still bistable and inconsistent when dropping. The stop point of the cradle was moved forward, which resolved the issue. Another problem arose when the original lift system lacked the torque to raise the cradle reliably. A simple servo-powered winch was introduced, providing sufficient lifting force without adding unnecessary complexity.

Additional upgrades included redesigned mounts for the battery and Raspberry Pi, which helped tidy the system's internal layout. The mount for the non-motorized caster was also redesigned to account for a new caster profile and corresponding change in height.

Overall, each round of testing yielded valuable insights. Every modification has contributed to a more robust and reliable system.

6.4. Web-App Testing and Improvements

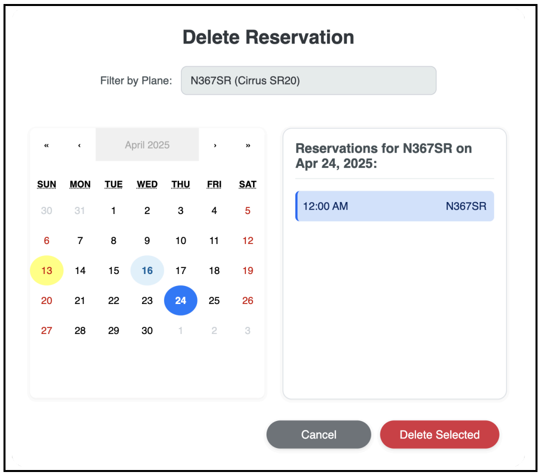
Different areas of the web app were improved after a holistic testing phase of its existing functionality. To accomplish an acceptable result, a P-diagram [6-6] with noise and target parameters has been implemented, as shown in Figure 6.4.1. Standard P-diagrams used for mechanical and system designs use quantitative parameters and measures. However, the user experience is qualitative in nature and should be evaluated as such. This requires that problems, targets, and new design parameters are related to the front end (user interface) and back end ("algorithm" and database) of the software. The main concerns of the testing were a bug where the system crashed and could not retrieve the list of reservations when a plane (with associated reservations) was deleted, an unclear UI interface for deleting reservations where a very simple dropdown menu was shown (to find a reservation could become hard with tens of rows), and finally the problem where the user could reserve a flight in a very short time-frame (5 minutes) which could have led to a general system management problem.



When a plane is deleted, all associated reservations should be also removed
Reservations must be made at least 10 hours in advance
Change in the menu for reservations
Changes in:
PlanesMgmt.js
Parameters
ReservationsMgmt.js
ScheduleFlightMenu.jsx
Redesign
No errors or warnings
Fast and continuous database processing flow (verified through MongoDB Compass)
Targets
User Dashboard Inputs
Product
Frontend – UI interface
Backend – Processing code and database
No frontend bugs
No backend bugs
No "real-life" implementation problems
Acceptable
Input Signals
Quality Measures
Unstable Wi-Fi
Slow database response
Slow computer
Noise

Figure 6.4.1: Web-App P-diagram

Figure 6.4.2 shows the new menu to delete reservations. In the top part, it is possible to filter by plane (or all planes). The calendar shows today's date in yellow, in light blue where there is at least one reservation, and in dark blue the selected date; on the right, a new menu window contains the reservations, which, when selected, can be deleted.



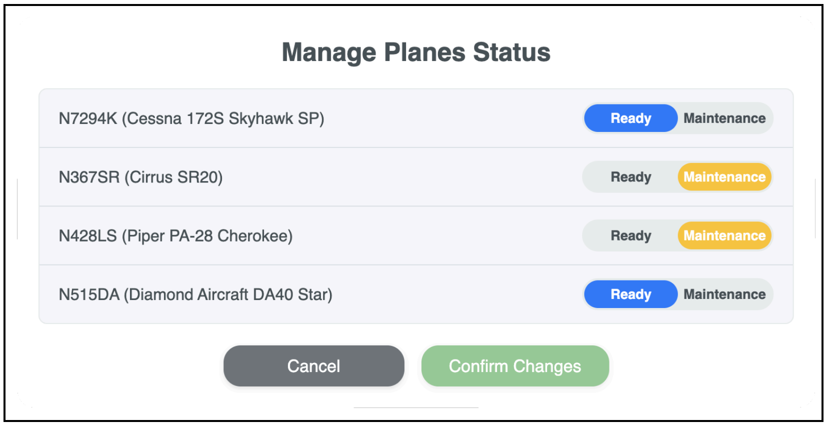
Delete Reservation

Filter by Plane: N367SR (Cirrus SR20)

Reservations for N367SR on Apr 24, 2025:

12:00 AM N367SR

Figure 6.4.2: New Menu for Deleting Planes

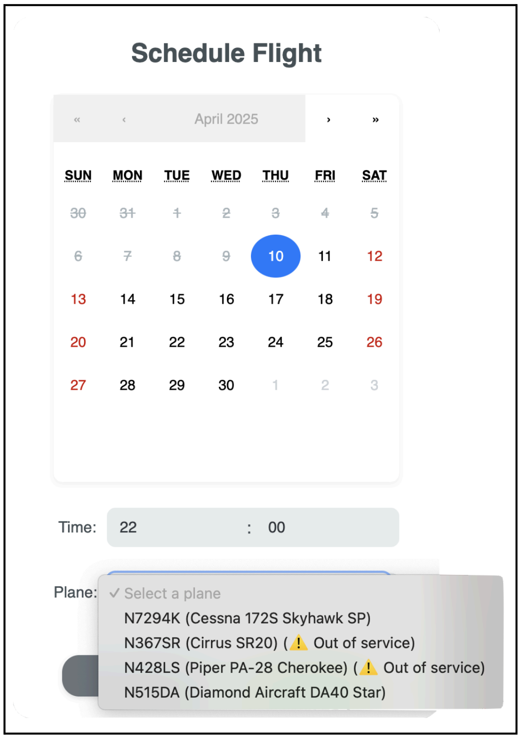


Manage Planes Status

N7294K (Cessna 172S Skyhawk SP)	Ready
N367SR (Cirrus SR20)	Maintenance
N428LS (Piper PA-28 Cherokee)	Maintenance
N515DA (Diamond Aircraft DA40 Star)	Ready

Figure 6.4.3: Menu to Schedule Maintenance

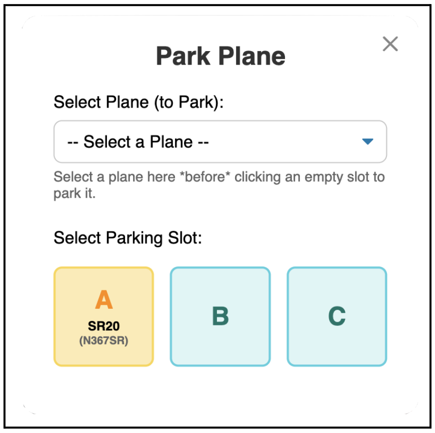
From the previous version, other functionalities have been added. Since it is now possible to schedule maintenance on planes (Figure 6.4.3), the calendar (Figure 6.4.4) now needs to show in the dropdown list a warning for planes out of service.



Schedule Flight

Time: 22 : 00

Plane:
- ✓ Select a plane
- N7294K (Cessna 172S Skyhawk SP)
- N367SR (Cirrus SR20) (⚠ Out of service)
- N428LS (Piper PA-28 Cherokee) (⚠ Out of service)
- N515DA (Diamond Aircraft DA40 Star)



Park Plane

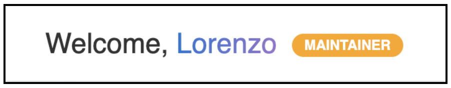
Select Plane (to Park): -- Select a Plane --

Select a plane here *before* clicking an empty slot to park it.

Select Parking Slot:

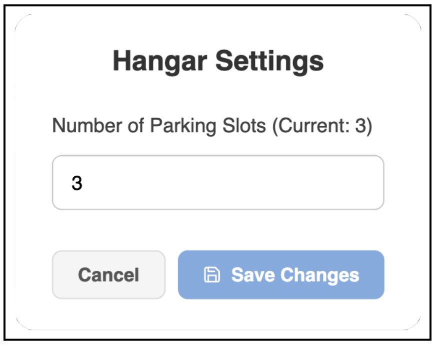
Figure 6.4.5: New Menu to Park Planes

Figure 6.4.4: New Menu to Schedule Flights



Welcome, Lorenzo MAINTAINER

Figure 6.4.8: Badge for Maintainer

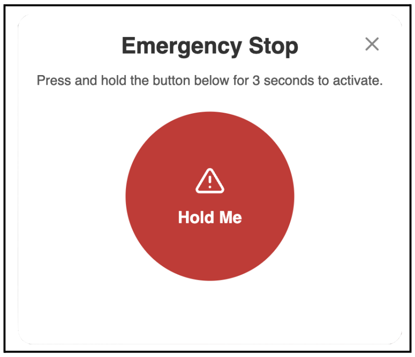


Hangar Settings

Number of Parking Slots (Current: 3)

3

Figure 6.4.6: Hangar Settings



Emergency Stop

Press and hold the button below for 3 seconds to activate.

Figure 6.4.7: Emergency Stop

Among the new functionalities, there is also a menu to park the plane after the flight (shown in Figure 6.4.5): After selecting the aircraft, it is possible to choose one of the parking slots identified by a letter. The number of slots can be changed in the new hangar settings menu, which is shown in Figure 6.4.6. The emergency stop button has been implemented (Figure 6.4.7), where a red button appears and has to be held for 3 seconds to activate the total shutdown of the system; the decision to implement a forced delay has been taken to ensure a voluntary activation by the user. Another UI feature that has been added is a maintainer badge that appears next to the name (Figure 6.4.8). The new functionalities needed additional Mongoose schema-based models. Table 6.4.1 shows the new hangar and parking slot schema, as well as the modified plane model, which now integrates the aircraft status.

Table 6.4.1: Mongoose Schema for new MongoDB Data

Schema	Properties			
Plane	N-number (*String*)	Brand (*String*)	Model (*String*)	Status *(String, enum)*
Hangar	Number of Parking Slots *(Number)*	Collection of *Parking Slots* (*ObjectId: Parking Slot*)		
Parking Slot	Number *(Number)*	1 *Plane* (*ObjectId: Plane*)		

6.5. Product Evaluation Summary

As the preceding sections have shown, the development due to the testing of the DUNG system of robots was significant. The embedded system was able to control a single robot according to the specifications required by the project requirements, following lines and recognizing April tags. In testing, this system performed well, where velocity and line following testing were conducted extensively to reveal a set of well-tuned PID controllers. Furthermore, testing the camera's reading ability of both blue tape and April tags pushed the team to experiment with shutter speeds and analog gain, resulting in the optimal settings for reducing motion blur. This robust recognition was aided by the incorporation of a ring light to illuminate both the tags and tape below, ensuring that variable lighting would not be a factor in the robot's performance. The improved lighting was added due to extensive camera testing. Lastly, new areas, indicated by green tape lines, were also implemented to slow the robots down before turning, increasing the reliability of reading the April tags.

Moreover, the path planning software was also streamlined, requiring fewer matrices and incorporating "rotation matrices" instead of "strafe matrices." The new matrices reflect a more efficient motion for the robots, as they will not have to contend with the mechanical limitations of sideways strafing and are more practical regarding the hangar layout that will be used in the demonstration.

The website was also given an overhaul on the user side. Users are able to park planes in different parking sites, give each plane a "ready" or "in maintenance" status, alter the dimensions of the hangar, and use a digital emergency stop button. The team discussed these features at length to improve the user experience. Moreover, these new features were also found through extensive testing of the website, as shown in the P-Diagram in the corresponding section.

In addition, the mechanical design of the robots was greatly improved. This is most clear when looking at the winch and camera mount, which improved the functionality of the lifting system and the camera vision system. On the other hand, parts that have been deemed suitable for the final robots were incrementally improved through testing. For example, the dimensions of the motorized caster mounts were improved based on testing to reduce wobble and improve the fit, which was then 3D printed.

7. Design Summary, Final Conclusions, Recommendations

7.1 Mechanical Design Summary

There are three main components of the system design: the frame, the movement, and the lifting. The initial design for the frame was maintained to the end. It utilizes aluminum extrusions like the one shown in Figure 7.1.1, arranged in a "U" shape. This allows the frame to be rigid, and the T-slot system allows for near infinite customization options and configurations. This was very helpful for testing, as it allowed us to quickly prototype and test different systems.

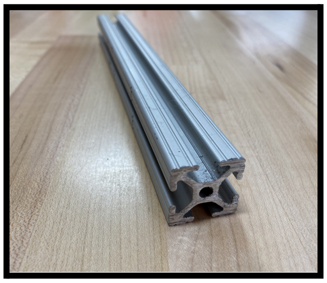


Figure 7.1.1: Aluminum Extrusion

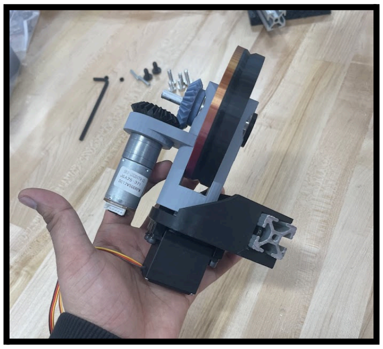


Figure 7.1.1: Drive Wheel System

The motorized caster design underwent some iterations, but the basic design stayed consistent. The fixed frame (pictured in black) mounts to the extrusion frame and supports a basic servo motor. This servo motor rotates the rotating mount (pictured in gray) from 0 to 180 degrees. The rotation was constrained to 180 degrees to reduce the complexity of the design. This meant that a slip ring was not necessary for this design. However, this reduced the efficacy of the holonomic motion of the system. On the rotating mount, a DC motor is mounted. It is mounted vertically to reduce the space it takes up when rotating. There is a set of 2 bevel gears that transmit the motion of the motor to the wheel. The wheel is supported on a 6mm D-slot axle, and 12mm x 28mm x 8mm bearings were used to reduce friction. There is one on either side of the wheel.

The cradle mechanism was also further improved, with some changes to the entryway ramp. Upon testing, the ramp had too much friction to allow the plane wheel to enter the cradle. Instead, the cradle was redesigned to have a "gate". On each side of the cradle, a micro servo was mounted to operate a rack and pinion system. This allows the front of the cradle to open, the plane wheel to slide in, and the front to close for lifting. For lifting, a winch system was implemented. A servo mounted in the middle of the frame operates a spool, which supports the plane wheel being lifted.

The final work on the robot was just to add an aesthetic electronics enclosure and clean up the wiring.

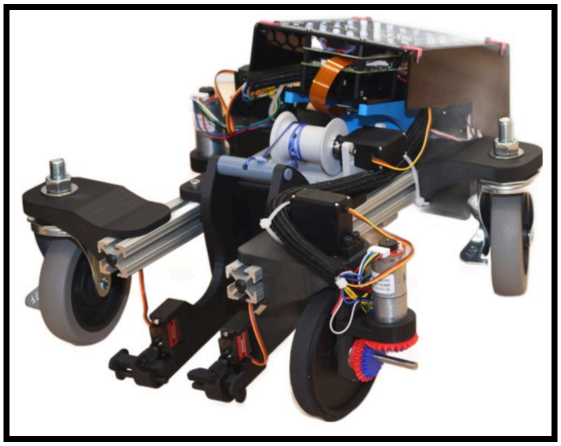


Figure 7.1.3: Finalized D.U.N.G. Beetle

On these robots, the electronics, bearings, axles, servos, motors, and casters can be reused. The aluminum extrusion can be disassembled and reused. The PLA and other plastic components can be shredded and recycled.

7.2 Embedded System Design Summary

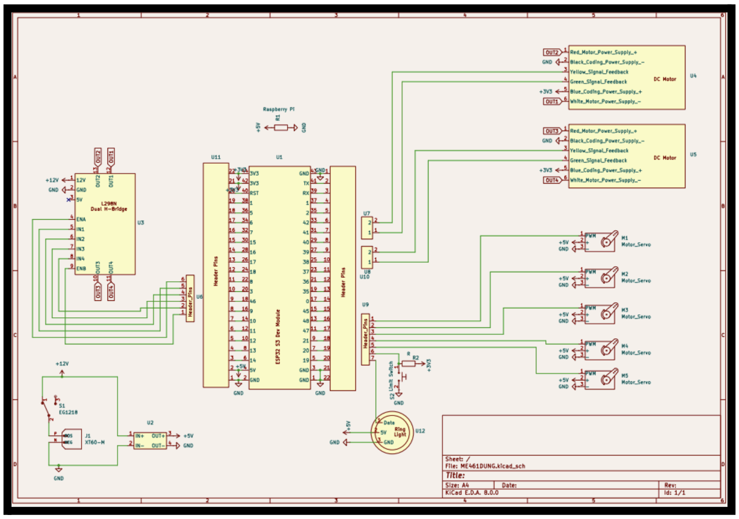


Figure 7.2.1: Electrical Schematic

The electronics are connected as follows: The 12V battery is connected to the motor driver and buck converter. The buck converter then powers the ESP-32 and the Raspberry Pi, where the former controls the servo motors and DC drive motors through the H-bridge and receives readings from the motor encoders. On the other hand, the Pi is connected to the camera and sends the video feed to a web server, which the processing computer can access. This computer then uses the information provided by the webstream to direct the robot, transmitting the information through the ESP32 connected to it to the robot's ESP32 as a 24-character message. The transmission is done through the ESP-NOW Bluetooth protocol. This process can be done for each robot individually or for all three robots in synchronicity. A schematic of these connections is shown in Figure 7.2.1, and a higher resolution image is shown in Appendix 7.2.1.

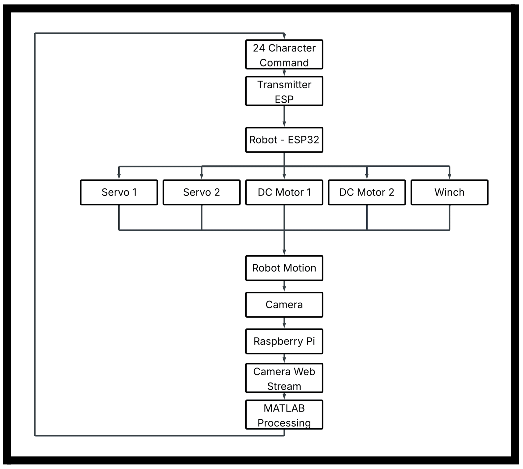


Figure 7.1.2: Control Loop Flowchart

There are two control loops in this system. The first is the control loop to manage wheel velocities on each robot. This loop is done individually on each robot's ESP-32, where encoder data is taken in, run through a Kalman filter, and used to find the speed of each wheel. Next, this data is used in a PID controller [6-2][6-3] to ensure each wheel reaches the target speed.

The second control loop manages the position of the robots relative to a line. This is done through the camera stream made by a robot's Raspberry Pi. Once the data is streamed out to the web server, a MATLAB script on the central processing computer analyzes the data and determines the robot's alignment with the line and its distance from the line. From here, the loop then uses a PD controller to adjust the robot's alignment by speeding up and slowing down the motorized wheels relative to each other. This PD controller is also used to adjust the servo motors' steering angle to move the robot's centroid closer to the center of the tape line. A diagram of this system is shown in Figure 7.1.2.

7.3 Map Layout Summary

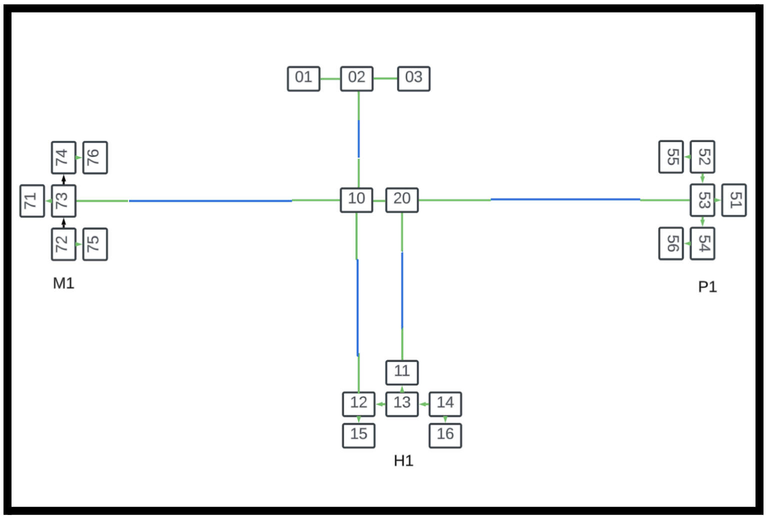


Figure 7.3.1: April Tag Layout with Color Coding

The hangar and surrounding area have a network of April tags and tape lines. The mock hangar has one hangar bay, a maintenance bay, and a parking bay. Lines connecting each April tag are laid out with tape and have different colors corresponding to robot speed. Blue lines indicate normal cruising speed, and green lines indicate instances where DUNG must travel slower to allow the camera to detect April tags. Each plane bay consists of six April tags: three for landing gear alignment and three as waypoints to such tags. Because the hangar bay is oriented to have the nose facing outward towards the door, April tag 20 was added so the nose wheel robot could reach April tag 11 at the correct orientation. This addition also maintains the center robot as the leader bot for synchronized motion. Notably, while blue lines can be as long as they must be, green lines are set to a minimum of one foot in length to ensure the onboard camera can read them.

As part of the hangar setup, a test plane was created to demonstrate the motion of the robots in a hangar with a direct analog to a full-size aircraft. This test plane was created using two aluminum poles arranged in a cross configuration. In addition, wheel assemblies for the mock plane were then designed in Onshape and 3D printed using a Bambu Labs X1C. This "plane" is shown in Figure 7.3.2. The resulting mock plane has enough clearance for any DUNG robot to maneuver underneath, and will ultimately serve as a constraint rig to hold the three robots together and to simulate the movement of a plane.

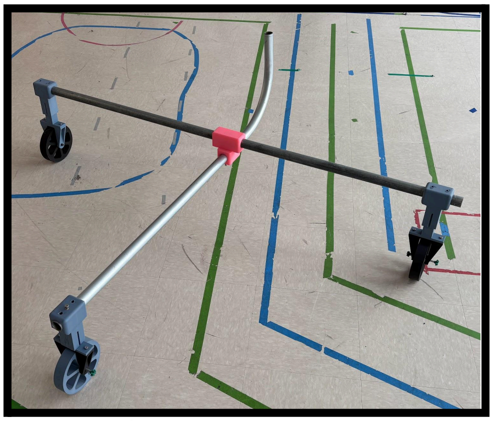


Figure 7.3.2: Mock Plane

Path planning is conducted by implementing the Floyd-Warshall algorithm, which computes the shortest paths between a set of nodes by iterating a "relationship matrix" that denotes the connections among all defined nodes. Once this relationship matrix is complete, the algorithm generates a path matrix, which denotes the next preferred node given the destination node specified by the user. The path-building code works by pulling values from the path matrix, constructing the path by changing the next "start" node until there are no possible nodes in between, upon which the path matrix returns a -1 value, denoting no shorter path.

Moreover, the relationship matrix denotes the cardinal directions of how each tag is positioned relative to another. This information in the relationship matrix can be used to map out two "junction matrices" between tags, quantifying each tag's relative positions. Then, the path logic will choose the direction the robot should travel, as indicated by matching values. Figure 7.3.1 shows that D1 represents the start junction, and D2 represents the end junction. Since the two values match up for a right-facing connection, the robot should rotate right, then travel forward. The ultimate direction command is chosen using a "rotation matrix," which plots out the relative direction the robot must traverse. This matrix is rotated with the robot's position, so if the robot performs a left rotation, the rotation matrix is rotated 90 degrees to the left.

7.4. Web-App Architecture and Data Model Structure Summary

The architecture chosen for this full-stack web application is a MERN stack (MongoDB, Express.js, React, and Node.js). As mentioned in Report 4 [7-1], a client-server model has been implemented where React is used for the front-end, Express.js and Node.js for the back-end, and API (which is the protocol to allow front and back ends to "communicate"), and MongoDB for the database (NoSQL type). These choices have been made to achieve an up-to-date, open-source, easily compatible, fast, cheap, smooth, and flexible web application.

A Mongoose schema-based model solution has been implemented to access the MongoDB database with an intuitive and fast approach. It is common to represent with an ERD (Entity-Relationship Diagram [7-2]) the overall structure and interactions of the models, where the numbers and symbols (0, 1, 0..1, *) represent the cardinality, that is how many instances of one entity relate to instances of another (1 = one, 0..1 = zero or one, * = zero or more). The ERD for the DUNG web app is shown in Figure 7.4.1.

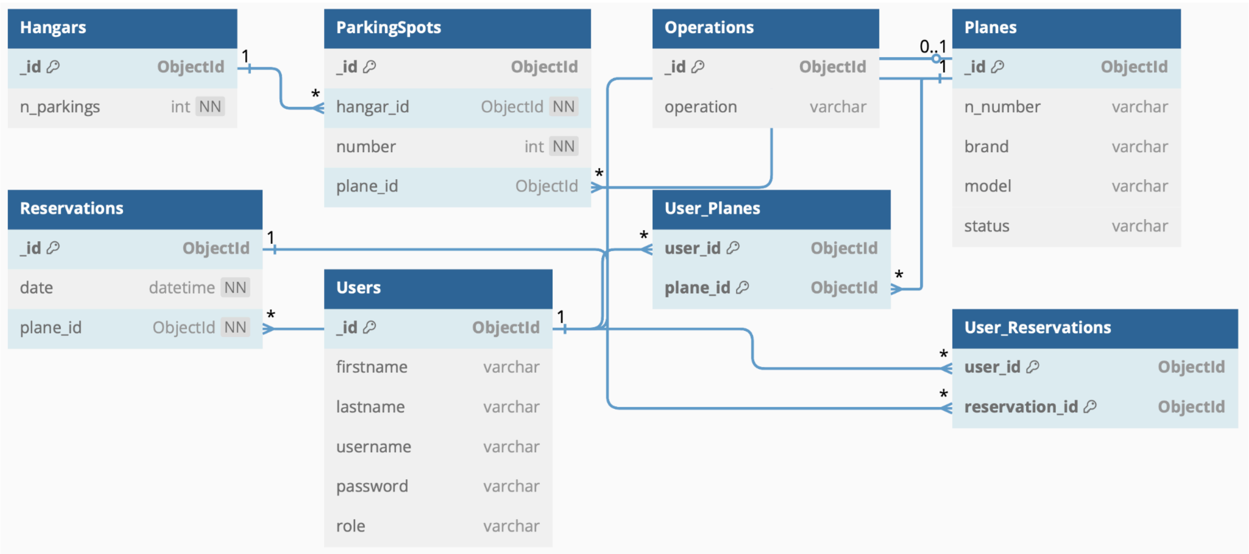


Figure 7.4.1: DUNG Entity-Relationship Diagram (ERD)

UI and UX Summary

DUNG's user experience has been designed to be as smooth and intuitive as possible. First of all, the user has to create an account with credentials that will allow access to the main dashboard shown in Figure 7.4.2. The user can either be a plane owner or a maintainer; in the latter case, the software grants access to all planes and reservations. Due to this elevated level of access, a security code is required during the registration phase. Once the user logs in, a dashboard will appear, which presents multiple options divided into three main categories: Planes to add, edit, or remove planes, Flights to plan and delete flights, and Others to park the plane after the flight or schedule a maintenance. In the lower section of the dashboard, the maintainer will also have the option to change the hangar's number of parking slots and to shut down the system in case an emergency occurs. Moreover, the top bar of the dashboard shows the name of the user (with a yellow badge in case the user is a maintainer), and a dedicated button to log out. The entire UI for the sub-menus is shown in detail in Appendix 7.

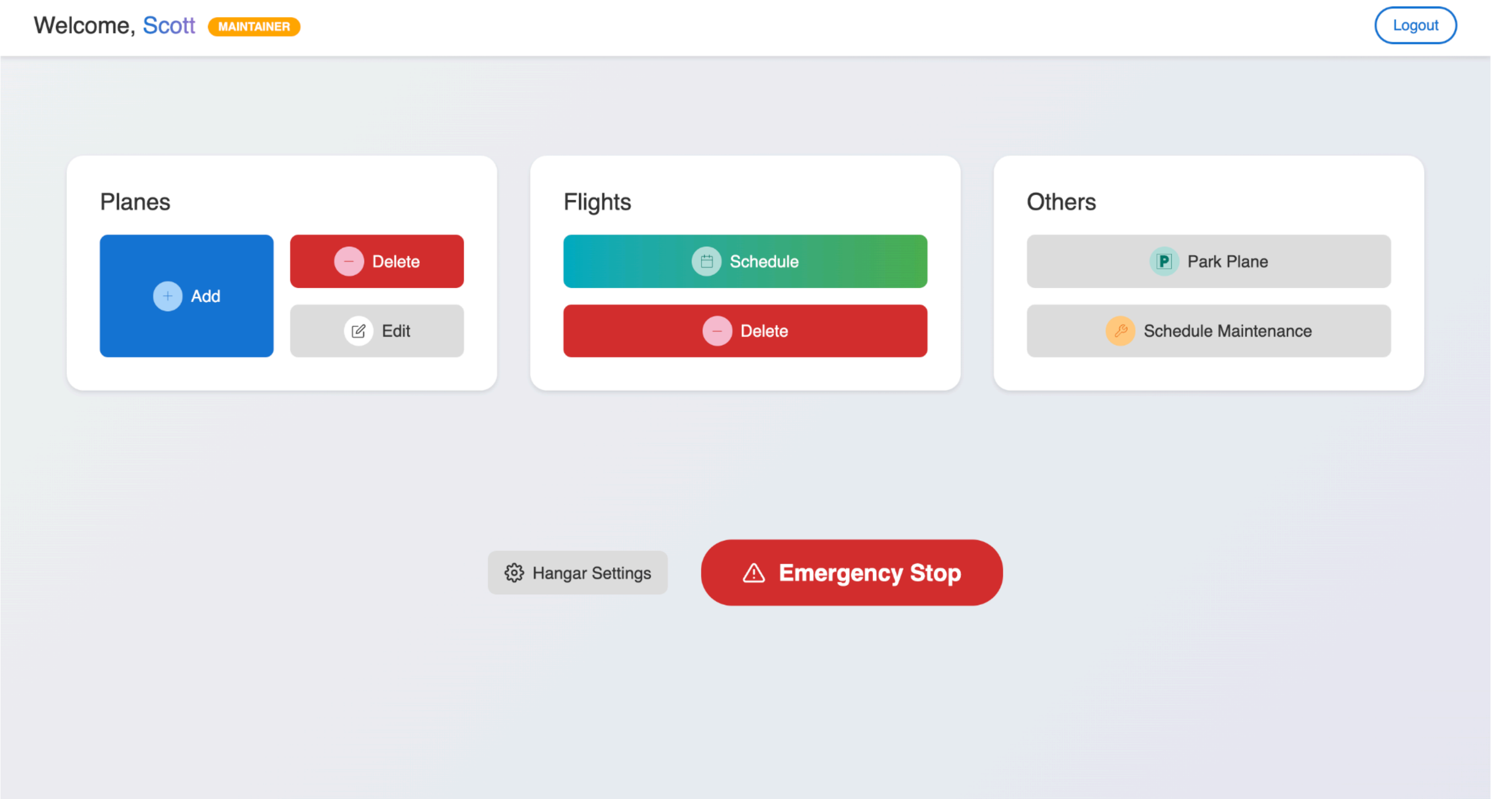


Figure 7.4.2: Main Dashboard UI

7.5. Overall Design Summary and Evaluation

There are a few key lessons learned that are consistent with all the subsystems, mainly in the domain of organization and project management. Firstly, it is clear that much of the time spent in the first semester on planning out the robots could have been used to build a prototype and allow the project to gain footing on the hardware front much faster. In addition, a clearer understanding of how each subsystem was meant to fit together in the project could have enhanced communication and collaboration and sped up the development of specific systems, such as the embedded system and the motion planning. Therefore, it may help future teams to have a more defined team structure, where a project manager ensures that the team cooperates fully and fast enough that the project will be completed on time.

DUNG has completed the construction and programming of all physical prototypes, and have been fine-tuned to ensure the entire system can go to the plane, latch onto it, and move it outside a mock hangar. All of this functionality has been built in its constituent parts and only required significant integration to be complete. This entailed testing all three robots while attached to the plane and ensuring that the back wheel robots maintain the formation centered around the front wheel robot. Moreover, the lifting and attachment systems were also optimized so that the entire system can operate autonomously and accomplish the goal of hangar management without being strenuous for a human user.

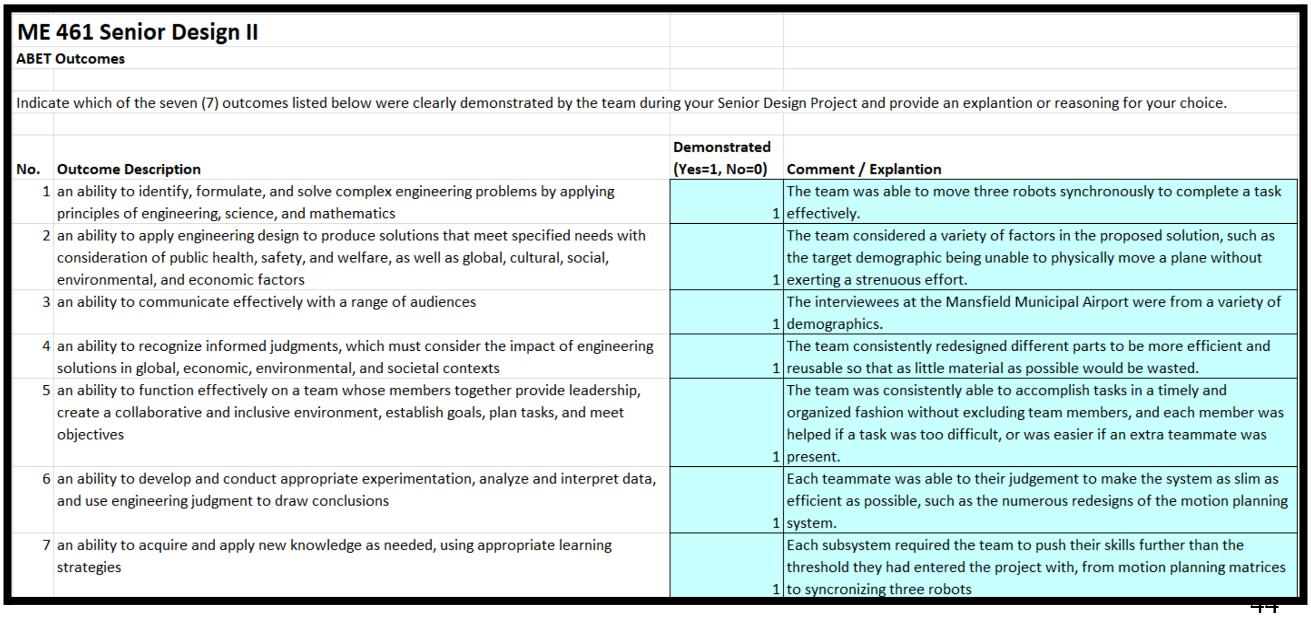
8. Engineering Ethics

Upon reviewing the NSPE code of engineering ethics, it is clear that the DUNG system of robots follows the given guidelines. Firstly, the robot is equipped with numerous safety commands, and an emergency stop which is always in view of the user via the web-app. In addition to this, the robots are consistently only in operation within their limits, which is achieved through speed limits, prescribed pathways for each robot, and when a user and maintainer have decided to move an aircraft. Moreover, the system is able to report issues, indicating where the issue has occurred, but this issue reporting system should be more detailed if this project is converted to a commercial product for client troubleshooting and other concerns. The robots also act solely to fulfill the duties given to them by users or maintainers, and have no way of acting without the will of the user. They are also unable to misrepresent their operational status, capabilities, sensor data or decision outcomes. Therefore, the DUNG system of robots upholds the honor of the engineering profession, maintaining a high standard of reliability, transparency and safety.

9. ABET Outcome

Upon completion of the DUNG project, the team was required to fill out the ABET accreditation form. This form was designed to assess the impact of the senior design process at Boston University, and the DUNG project specifically on the team. When filling out the form, it was clear to see that the team had been exposed to a litany of new ideas and design philosophies, which were used to solve complex engineering problems. In addition to this, after conducting the interviews at Mansfield Municipal Airport in September 2024, at every junction, the team considered the impact of their decisions on the client needs, putting a name behind what would have otherwise been a faceless list of targets. Thus, every ABET outcome had been achieved, which is reflected in Table 9.1.

Table 9.1: ABET Accreditation Worksheet



ME 461 Senior Design II

ABET Outcomes

Indicate which of the seven (7) outcomes listed below were clearly demonstrated by the team during your Senior Design Project and provide an explantion or reasoning for your choice.

No.	**Outcome Description**	**Demonstrated (Yes=1, No=0)**	**Comment / Explantion**
1	an ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics	1	The team was able to move three robots synchronously to complete a task effectively.
2	an ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors	1	The team considered a variety of factors in the proposed solution, such as the target demographic being unable to physically move a plane without exerting a strenuous effort.
3	an ability to communicate effectively with a range of audiences	1	The interviewees at the Mansfield Municipal Airport were from a variety of demographics.
4	an ability to recognize informed judgments, which must consider the impact of engineering solutions in global, economic, environmental, and societal contexts	1	The team consistently redesigned different parts to be more efficient and reusable so that as little material as possible would be wasted.
5	an ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives	1	The team was consistently able to accomplish tasks in a timely and organized fashion without excluding team members, and each member was helped if a task was too difficult, or was easier if an extra teammate was present.
6	an ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions	1	Each teammate was able to their judgement to make the system as slim as efficient as possible, such as the numerous redesigns of the motion planning system.
7	an ability to acquire and apply new knowledge as needed, using appropriate learning strategies	1	Each subsystem required the team to push their skills further than the threshold they had entered the project with, from motion planning matrices to syncronizing three robots

References:

[2-1] Iamtheari. "Marking Hangar Floor." Pilots of America, Pilots of America, 11 June 2022, www.pilotsofamerica.com/community/threads/marking-hangar-floor.138436/.
[2-2] "Beechcraft Owners & Pilots Group Web Forums." BeechTalk, www.beechtalk.com/forums/index.php. Accessed 30 Sept. 2024.
[2-3] Durden, Rick. "Aircraft Tugs: Electric Power Dominates." Aviation Consumer, 30 Dec. 2019, www.aviationconsumer.com/industry-news/editorial/aircraft-tugs-electric-power-dominates/.
[2-4] "Aircraftcaddy 4K Jr.: Battery-Powered Aircraft Tug." DJ Products, 12 Aug. 2024, www.djproducts.com/product/aircraftcaddy-4k-jr/.
[2-5] AC AIR Technology. "Compact Tugs." AC AIR Technology, acairtechnology.com/collections/compact-tugs/carbon-cub. Accessed 30 Sept. 2024.
[2-6] "Quality Function Deployment Template." Quality Function Deployment Template – Continuous Improvement Toolkit, citoolkit.com/templates/quality-function-deployment-template/. Accessed 30 Sept. 2024.
[2-7] T. Ueyama, T. Fukuda, and F. Arai, \Conguration of communication structure for distributed intelligent robot system," in Proc. IEEE Int. Conf. on Robotics and Automation
[2-8] G. Dudek et al., \A taxonomy for swarm robots," in IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, (Yokohama, Japan), July 1993.
[2-9] M. Mataric, \Minimizing complexity in controlling a mobile robot population," in IEEE Int. Conf. on Robotics and Automation, (Nice, France), May 1992
[2-10] Woodman, Oliver J.. "An introduction to inertial navigation." (2007).
[2-11] Diebel, James. "Representing Attitude : Euler Angles , Unit Quaternions , and Rotation Vectors." (2006).

[5-1] Material Properties of T-Slot Aluminum Extrusions: https://rmmc.net/8020/technical-data/
[5-2] Material Properties of PLA: https://bambulab.com/en/filament-guide
[5-3] ESP32-Properties: https://docs.espressif.com/projects/esp-dev-kits/en/latest/esp32s3/esp-dev-kits-en-master-esp32s3.pdf
[5-4] Bridgeport: https://asia.bridgeportmachinetools.com/wp-content/uploads/sites/11/1330J_BridgeportSeriesI_2023.pdf
[5-5] Floyd Warshall Algorithm: https://www.geeksforgeeks.org/floyd-warshall-algorithm-dp-16/
[5-6] Floyd Warshall Implementation: https://www.mathworks.com/matlabcentral/fileexchange/11549-floyd-shortest-path-routing

[6-1] M. Higgins, A. Keller, Z. Pinard, G. Rigamonti, "Project 118: Viewmaster for Friction Visualization Final Report", Boston, MA, USA: ME461 Capstone Experience, May 2024.

[6-2] Li, Yun, et al. “PID Control System Analysis and Design.” IEEE Control Systems Magazine, vol. 26, no. 1, 2006, pp. 32–41, eprints.gla.ac.uk/3817/1/IEEE3.pdf, https://doi.org/10.1109/mcs.2006.1580152.

[6-3] Zhang, Hao Ming, et al. “Servo Motor Control System and Method of Auto-Detection of Types of Servo Motors.” Applied Mechanics and Materials, vol. 496-500, Jan. 2014, pp. 1510–1515, https://doi.org/10.4028/www.scientific.net/amm.496-500.1510.

[6-4] Joyce, and Cornelis Teunissen. Perceived Sharpness in Moving Images. 1 Oct. 1990, https://doi.org/10.1117/12.19662. Accessed 3 May 2023.

[6-5] Grgić, Ivan, et al. “Accuracy of FDM PLA Polymer 3D Printing Technology Based on Tolerance Fields.” Processes, vol. 11, no. 10, 1 Oct. 2023, p. 2810, www.mdpi.com/2227-9717/11/10/2810#:~:text=In%20the%20same%20paper%2C%20their, https://doi.org/10.3390/pr11102810.

[6-6] D. G. Ullman, The Mechanical Design Process, 4th ed. New York, NY, USA: McGraw-Hill, 2010, ch. 10, pp. [281-284]

[7-1] A. Singh, L. Barale, M. Sukkar, Y. Alhajji, Z. Wu, and Z. Bhatti, TEAM 183 Report 4 for Submission, Boston University, Boston, MA, USA, 2024.

[7-2] Lucidchart, "ER Diagrams: A Complete Guide," Lucidchart. [Online]. Available: https://www.lucidchart.com/pages/er-diagrams. Accessed: 25-Apr-2025.

Appendix 3: Customer Interview Notes

KENN SEBESTA

- Get plane read →moved
- Lots of variance
- Friend Chris takes seconds to pull plane out
- Sebesta 2 minutes just to take it out
- Took hangar mate "quite some time"
- (extreme case) some people (even with powered dolly) couldn't do it (human error, misalignment)
- Small plane individual hangars (luxury option)
- Sometimes must call 1 hr ahead to retrieve airplane
- Typically airplanes flown once a month
- Juggling unlikely (getting multiple planes out/back in)
- On the spur calls of who want to fly are possible
- If working on "bob"s airplane and you have to get it out the back of hangar
- Biggest gripe is propensity for hangar rash: very delicate and cannot bump into anything. Fragile.
- Heavy (1400-1800lb) 2300lb at the high end
- Carrying speed: not fast or far but still heavy
- Hangar lips can be challenging (even as small as 5mm)
- Long wings (front end swing) watch out for lever effect
  - Margin for pilot error
- Hangar door size 40ft (small end 40ft large end 48ft)
- 40ft door is pretty standard
- 60,70,80ft doors for community hangar
- Max plane amount per hangar: dozens of gliders, 10 light singles per community hangar
- Larger hangars out west (10 private jets)
- Conventional (tail wheel), rarer, tricycle (three wheel front), "nosedragger"
- Community hangars are desired
- Constrictions on what times you can fetch your aircraft
- T hangar: 40 ft door and standard depth
- Similar tech: RC cars that move helicopters
- Pitfalls of system: airplane damage, be resource/utilization practical
- Each airport runs differently, here land is either owned by city or condo structure
- "Standard": airport manager gets to set price/division of resources
- No sparks! Aircraft regulations
- No leaving airplanes in front of hangars when shuffling around.
- Expected time: "slow is fine"
- 1hr is an extreme outlier to get one airplane out
- If 10 airplanes, 3 min per airplane
- Leave airplane somewhere that it doesn't block anyone else
- Keep hangar door closed when possible
- Fuselage should remain dry
- Mistake from all senior design classes: get data now instead of later.
- Make robot concept + large design concept (rough prototype)
- "Hardware is hard"
- Battery power is more than enough to move small distances
- Cottage industry for tool battery packs (drill batteries)
- Add emergency stop (deadman switch)

- Hangar rash (move damage)

STEVE

- Plane is 4000 lb, difficult to move solo
- Factory tow bar has no leverage (connects to front wheel)
  - Highly inconvenient
- Modified milwaukee drill with a chain
- Diamond pattern wheel will turn the front landing gear
- Battery lasts very short: 4 movements (taking it out/putting it back into hangar)
- Difficult to move the plane upward
- In rain and winter no friction, can’t use the gear tool
- Uses winch to move plane back in
- Turning circle is not a big problem putting planes away.
- Minimal tire wear with diamond tool, landing wears off more than the tool.
- Normal maintenance unaffected.
- Aviation consumer reports: testing shows wear is negligible.
- In germany, hangars are roundabout push button to get plane
- Planes have individual turn limits (restrictions vary)
- Gear can be damaged if gear is overturned
- Cost of manual motor: $1800-$2000
  - Mainly for larger extension arm
- Device able to be stowed into nose compartment
- Non universal nose gear

THOMAS

- Hangars must be built from the ground up
- Has enough slope that there is little force required to remove the plane
- Steps to remove:
  - Place mats down for hangar lip
  - Remove chocks
  - Attach arm
  - Pull
- Damage primary concern
- Aircraft alignment is paramount (many reference points)
- Four possible dangers
- RC Cars exist
- To put back in
  - Align centerline with visual aid
  - Constantly checking reference
  - Push in
- Small changes with steering have large effects on wingtips
- Do not correct on the fly. Start over.
- Vehicle dynamics dictate large changes down the line
- Uses simple tow bar
- 3 to 4000 for RC car
- 50 to 100 bucks for tow bar
- Also has automated drill powered mover
- $1500

- Chain driven at its own wheel
- Takes longer to setup
- Lifts up and puts down machine to turn airplane
- Significantly less traction
- 1600lb aircraft
- Device is very poor on rain and snow
- Front is coupled to rudder
- 10-20 times can use automated arm
- The further in your aircraft is the more consequential
- Glider has 3 crosshairs for alignment
- Glider is held using a wheel cradle similar to car cradles
- Car cradles are a lot wider than airplane ones
- Must know where aircraft is going to go (good planning)
- Utilize floor space efficiently (have safety zone added to aircraft)
- “If you are part of a community hangar you are screwed”
- Hangar round table design is common in Europe (small hangar opening)
- Can stack high wing and low wing
- “What is your hangar layout? My algorithm will tell you the best way to take your aircraft out.”
- “Oh shit, I haven’t thought about this” -aussie knot
- Target group: fixed base operator
- Companies at larger airports will provide hangar space

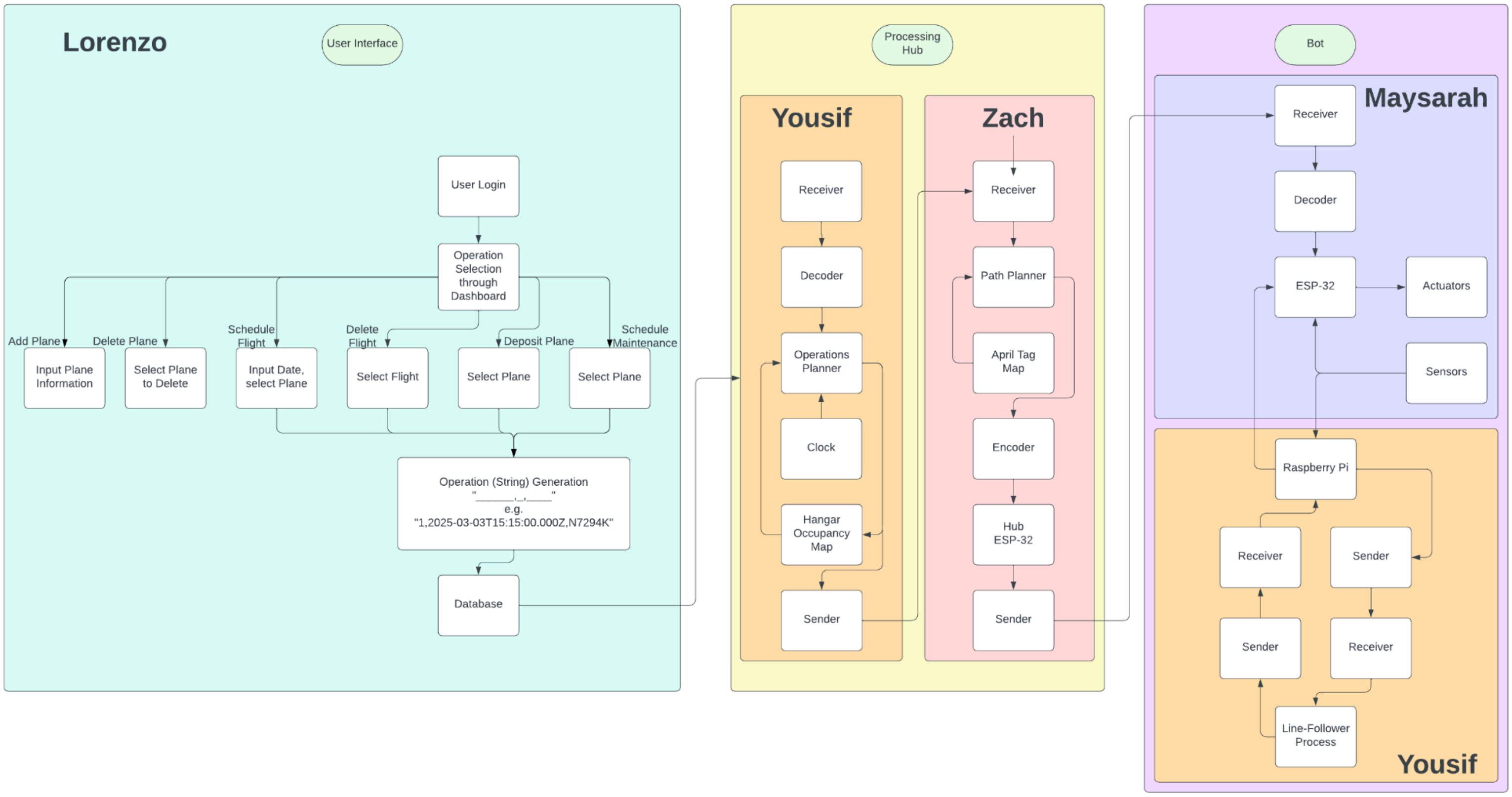
JOHN

- Has gas and electric mover
- Electric is easier (built in charger)
- Can be adjusted higher or lower
- As fast as you can walk to move it
- Not very involved process
- 2600 bucks for mover electric 1, 4000 for gas one, 3000 for other electric one
- Has manual bar for travelling
- Maximum 3 people: 1 moving 2 watching
- RC car option available
- Hangar door is manual, has a kill plug for emergencies.

Appendix 4: Brainstorming Session Drawings

See the following shared folder for all brainstorming session images. https://drive.google.com/drive/folders/1-2q7CsnVsw3QBUsGhmpbaYzPOo_vAoC6?usp=sharing

Appendix 5: Product Generation



Appendix 5.3.1:Higher Resolution Process Flowchart

Appendix 7: Design Summary

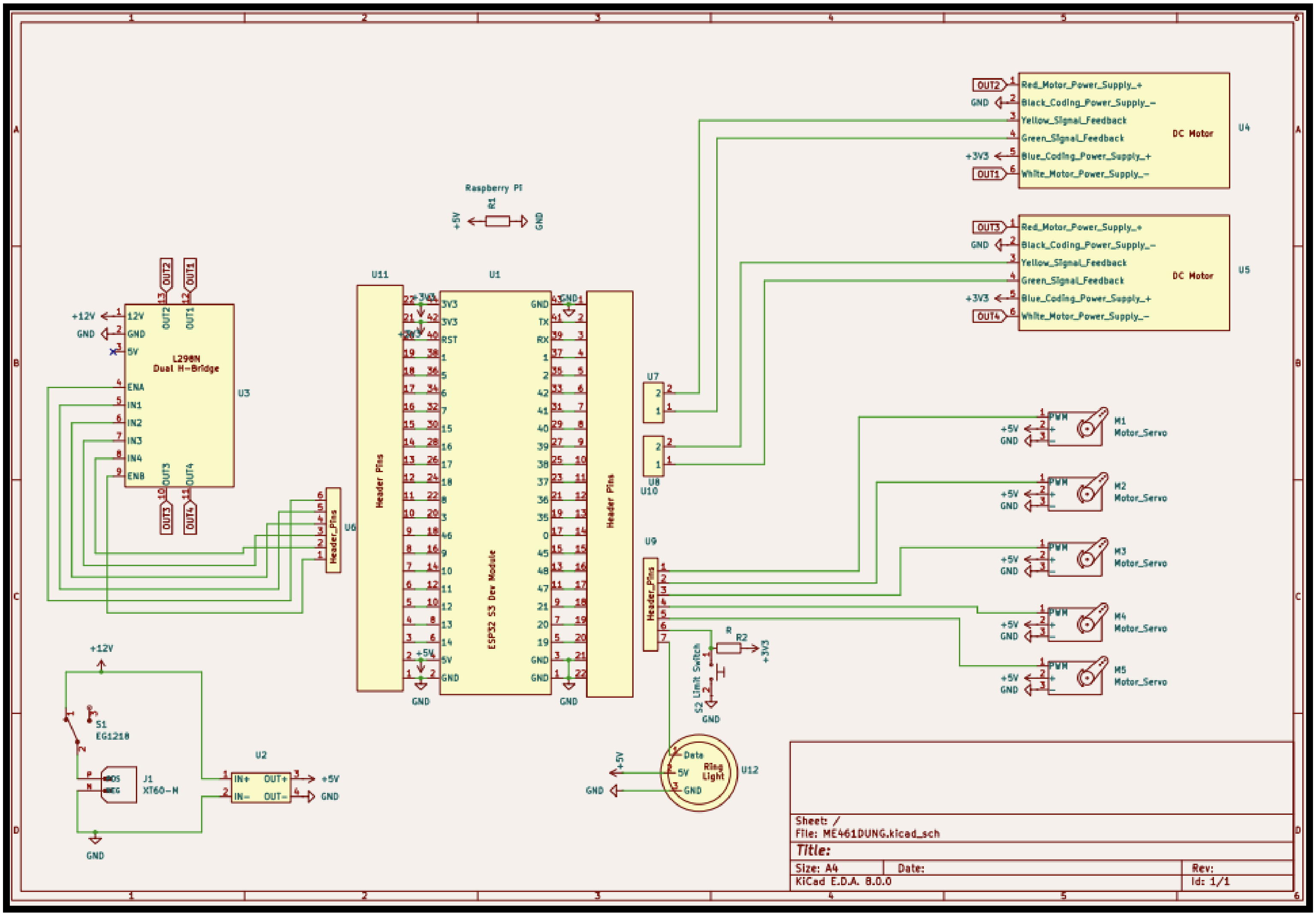
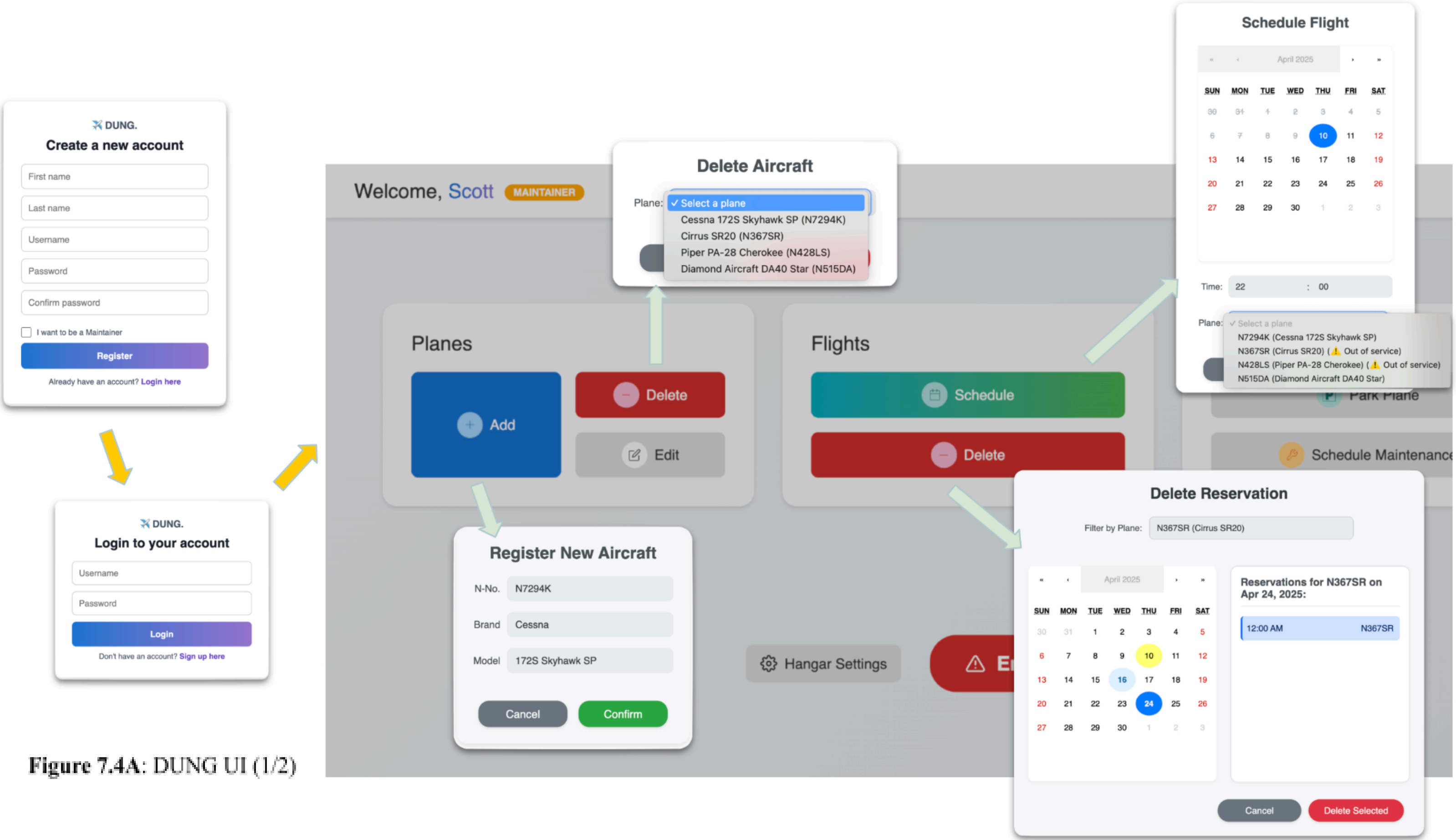


Figure 7.2.1: Higher Resolution Electrical Schematic


DUNG.
Create a new account
First name
Last name
Username
Password
Confirm password
□ I want to be a Maintainer
Register
Already have an account? Login here
DUNG.
Login to your account
Username
Password
Login
Don't have an account? Sign up here
Welcome, Scott MAINTAINER
Planes
Add
Delete
Edit
Flights
Schedule
Delete
Hangar Settings
Delete Aircraft
Plane:
✓ Select a plane
Cessna 172S Skyhawk SP (N7294K)
Cirrus SR20 (N367SR)
Piper PA-28 Cherokee (N428LS)
Diamond Aircraft DA40 Star (N515DA)
Register New Aircraft
Cancel
Confirm
Schedule Flight
April 2025
SUN MON TUE WED THU FRI SAT
Time:
Plane:
✓ Select a plane
N7294K (Cessna 172S Skyhawk SP)
N367SR (Cirrus SR20) (⚠ Out of service)
N428LS (Piper PA-28 Cherokee) (⚠ Out of service)
N515DA (Diamond Aircraft DA40 Star)
Park Plane
Schedule Maintenance
Delete Reservation
April 2025
SUN MON TUE WED THU FRI SAT
Cancel
Delete Selected

N-No. N7294K

Brand Cessna

Model 172S Skyhawk SP

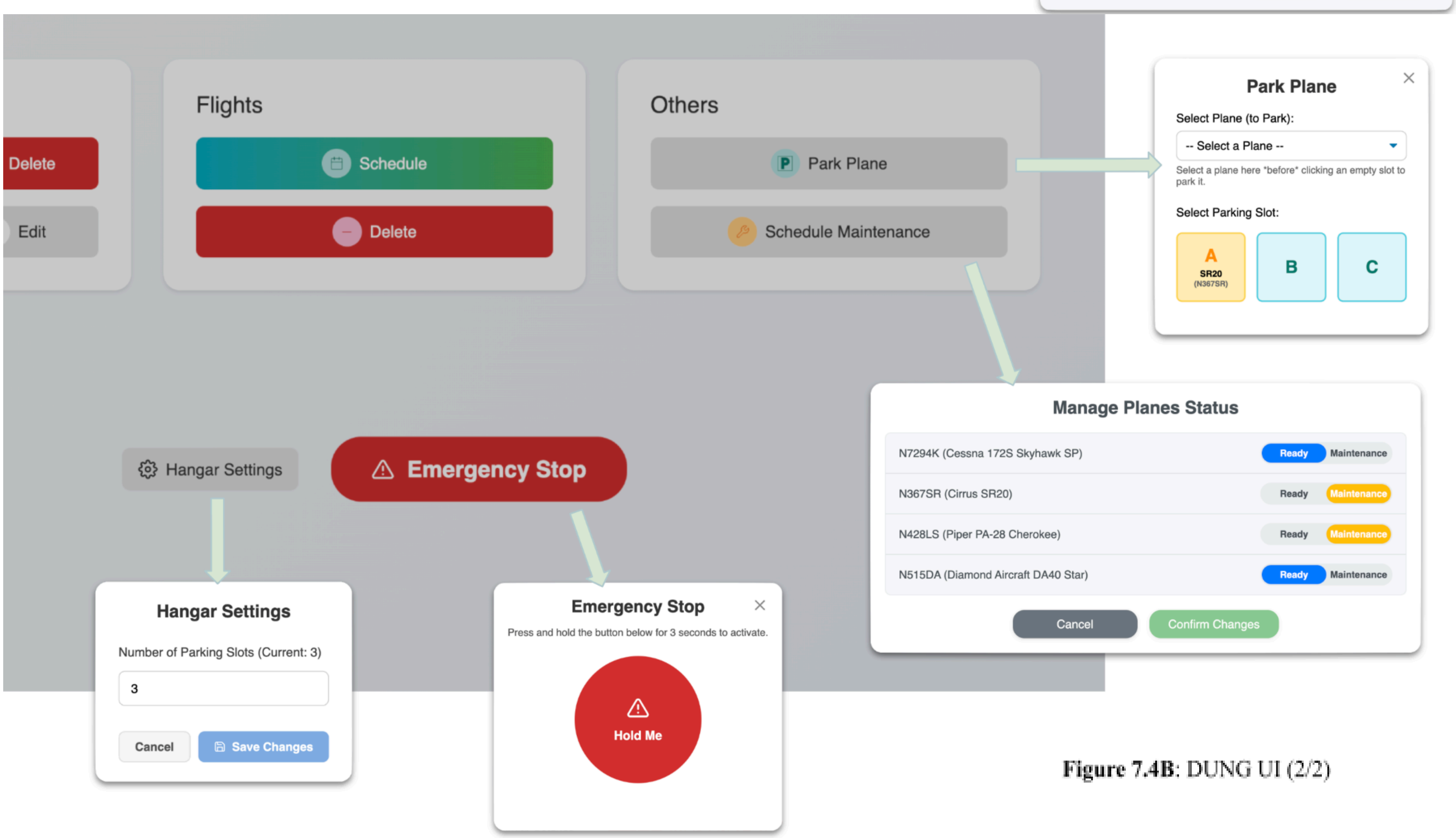
Time: 22 : 00

Filter by Plane: N367SR (Cirrus SR20)

Reservations for N367SR on Apr 24, 2025:

12:00 AM N367SR

Figure 7.4A: DUNG UI (1/2)



Select Plane (to Park): -- Select a Plane --

Select a plane here *before* clicking an empty slot to park it.

Select Parking Slot:

N7294K (Cessna 172S Skyhawk SP)	Ready
N367SR (Cirrus SR20)	Maintenance
N428LS (Piper PA-28 Cherokee)	Maintenance
N515DA (Diamond Aircraft DA40 Star)	Ready

Number of Parking Slots (Current: 3) 3

Press and hold the button below for 3 seconds to activate.

Figure 7.4B: DUNG UI (2/2)