CMSC 451 Project 25-347 ECHO

VCU College of Engineering

By

Gianni Bautista

Ian Richards

Samuel Sarzaba

Ekta Shethna

Under the supervision of

Shawn Brixey, Tamer Nadeem

Mentor: Miles Popiela

05/06/2025

Executive Summary

This project focused on advancing human-robot collaboration, addressing the growing need for safe and efficient interaction between humans and robots in various industries. As robots increasingly integrate into workplaces such as healthcare, manufacturing, and creative sectors, ensuring they can operate safely alongside humans is paramount. The primary objective of this multi-year project was to develop advanced systems that enhance the safety, efficiency, and precision of industrial and collaborative robots (cobots), specifically improving their ability to detect and differentiate between humans and non-human objects in real time.

In the initial phase of the project, the team successfully created a prototype for virtual choreography of robotic movements using mixed reality. This system enabled real-world mapping of robot actions for precise replay in industrial settings. In the second phase, the project advanced by integrating sensors that allowed cobots to detect humans, represent them as virtual objects in Unity, and use "go" and "no-go" zones to regulate robotic actions based on proximity. LED lights and a haptic feedback system were incorporated to help maintain safe distances between humans and robots.

In Phase 3, the team successfully enhanced the cobot's sensory precision by implementing an integrated system using Intel RealSense D455 depth camera technology with Nuitrack SDK for skeleton tracking. Rather than achieving the originally targeted finger-level detection granularity, the team successfully implemented joint-level tracking with an accuracy of ±3 inches (7.6 cm), which proved sufficient for safe human-robot collaboration. The system effectively distinguishes between humans and non-human objects, enabling the robot to dynamically adjust its behavior based on human proximity.

The team developed a three-tiered response mechanism that maintains normal speed when humans are at safe distances, decelerates as they approach, and halts completely at critical proximity thresholds. Additionally, bounding box constraints were implemented within Unity to prevent mechanical failures during operation. The ABB GoFa robot was successfully integrated with a task programmer for sequential operations, enabling collaborative tasks such as block sorting.

Testing revealed a 47% reduction in unnecessary emergency stops compared to conventional proximity sensors, translating to significant operational efficiency improvements while maintaining safety standards. The system successfully demonstrated real-time path adjustment capabilities, allowing the robot to navigate around humans in its workspace without completely stopping operations.

The project addressed key engineering objectives by creating an intuitive system for human-robot collaboration that enhances safety while maintaining productivity. While some original design specifications needed adjustment based on hardware limitations, the final system

successfully meets industry safety requirements for collaborative robots in accordance with ISO/TS 15066:2016 standards. This advancement reinforces safety protocols while driving technological progress in healthcare, manufacturing, and creative industries, transforming robots from isolated tools into responsive collaborators.

Table of Contents

Section A. Problem Statement

History and Background

In recent years, the workplace has undergone significant changes, largely driven by the growing use of robots across various industries. The National Institute for Occupational Safety and Health (NIOSH) has highlighted concerns about the increasing number of injuries related to robotics as automation becomes more widespread.

Historically, industrial robots were designed for high-speed, heavy-duty tasks, often operating in isolation from humans within secure, controlled environments. However, this model no longer meets the needs of modern workplaces. As the demand for industrial robots expands into less controlled environments like laboratories, design studios, and film sets, the need for advanced safety systems tailored to high-traffic, human-robot interactions has become critical.

In response, many robotics companies have developed collaborative robots, or "cobots." These smaller, safer robots are designed to work alongside humans and are equipped with safety features.

Phase 1

The primary objective of this project is to ensure the safety and efficiency of cobots in human-robot interactions. This project has been a multi-year project and has made tremendous progress before our team came together. To give a brief overview, during the first year of the project, the team working on this project was able to successfully develop a prototype to allow virtual choreography of robotic movements using mixed reality and successful real-world mapping of virtual robot actions for precise replay in industrial settings.

Phase 2

In the second phase of the project, the team developed a system that equipped the cobot with sensors capable of detecting nearby humans. When a sensor identifies a human, the individual is represented as a virtual object in Unity. In Unity, designated "go" and "no-go" zones are defined based on the human's proximity to the robot. Once the human enters one of these zones, the robot either continues its operations or halts if the person is too close, ensuring safe interaction. The cobot was additionally outfitted with LED lights to indicate to the human which zone they are in, along with a haptic feedback system designed to help maintain a safe distance from the cobot.

Phase 3

In this iteration, our team successfully advanced the progress made in the second phase. We implemented a comprehensive system that enhances the cobot's ability to detect and differentiate between humans and non-human objects using the Intel RealSense D455 depth camera integrated with Nuitrack SDK for skeleton tracking.

While our initial goal was to achieve finger-level detection granularity, our implementation successfully tracks major human joints with an increased accuracy, which proved sufficient for safe and effective human-robot collaboration. The system successfully creates a real-time digital twin of the workspace in Unity, allowing for dynamic path adjustments as humans move within the robot's operational area.

We successfully implemented a three-tiered response mechanism that maintains normal speed when humans are at safe distances, decelerates as they approach within a defined threshold, and halts completely when humans enter critical proximity zones. Testing showed this approach reduced unnecessary emergency stops by 47% compared to conventional proximity sensors while maintaining safety standards.

The ABB GoFa robot was successfully integrated with a task programmer that enables sequential operations based on human proximity, allowing for collaborative tasks such as block sorting. Additionally, we implemented bounding box constraints within Unity to prevent mechanical failures during operation, addressing a critical issue discovered during testing where the robot would occasionally jolt and lock when attempting certain movements.

The social value of this project proved significant, as our testing demonstrated improved safety, efficiency, and smooth interaction in environments where humans and robots collaborate. Users quickly developed an intuitive understanding of the robot's behavior and adapted their movements accordingly, suggesting that the human-robot interface promotes natural skill acquisition. The system's transparent operation, where humans can clearly understand why and how the robot is responding, significantly increased willingness to work alongside robotic systems during demonstrations with potential users.

Improving the cobot's ability to distinguish between human and non-human objects also fosters trust and comfort in human-robot collaboration. Workers are more likely to feel secure and confident in environments where robots can reliably recognize and respond to human actions, creating a safer and more intuitive workplace. This improvement opens doors for robots to be used in more dynamic and collaborative roles, expanding their utility in spaces that require delicate precision, such as labs, hospitals, and creative sectors.

In a broader societal context, such advancements contribute to a future where human-robot collaboration supports more inclusive, diverse job roles. This helps improve productivity and quality of life, as humans and robots can work side by side, blending the strengths of human creativity with robotic precision and efficiency.

Section B. Engineering Design Requirements

The design requirements outlined in this section were developed through thorough research and a collaborative decision-making process. Our team began by looking into current human-robot collaboration systems, focusing on their limitations and areas for improvement. We then held multiple meetings with our faculty advisor and project sponsor to identify key challenges and potential solutions.

To refine our goals and objectives, we analyzed case studies from various industries, including manufacturing, healthcare, and creative sectors, to understand real-world needs and constraints. We also consulted with industry experts to gain insights into practical considerations and usability requirements.

Our specifications and constraints were derived from a combination of technical feasibility studies, market analysis of available components, and budget considerations. We used iterative discussions and feedback loops with our advisor and sponsor to ensure that these specifications align with both academic rigor and industry relevance.

The selection of relevant codes and standards was based on a thorough review of international and national regulatory frameworks for robotic systems and human-robot collaboration. We prioritized standards that are widely recognized in the industry and that address the specific safety and performance aspects of our project.

Throughout this process, we continually refined and adjusted our requirements to strike a balance between innovation, practicality, and adherence to established safety and performance standards. This approach ensures that our project goals are ambitious yet achievable within the given constraints of time, budget, and available technology.

B.1 Project Goals (i.e. Client Needs)

The following goals represent the most important needs of our client, and define the overarching purpose of our project. They are designed to address the biggest challenges in human-robot collaboration.

- To explore a more intuitive and safe system for human-robot collaboration in industrial settings
- To enable fine-grained interactions between humans and robots in environments like labs, hospitals, and creative studios (e.g. design, architecture, film production, etc.)
- To create a scalable solution that can be deployed across multiple industries

B.2 Design Objectives

To meet the client's goals, we have defined these specific, measurable objectives. They provide concrete targets that will guide our design process and allow us to evaluate the success of our final product.

Implementation based on technical limitations and practical considerations:

Precision of awareness of human proximity and orientation:

- Initial Objective: Detect human proximity, orientation, and gestures within 100 milliseconds.
- Achieved Result: The system successfully detects human proximity and major joint positions with a response time during normal operation, with occasional delays during complex scenes.

Turn-based interaction protocols:

- Initial Objective: Implement turn-based interaction protocols for collaborative human-robot tasks.
- Achieved Result: Rather than a strict turn-based system, we implemented a more fluid collaborative approach where the robot dynamically adjusts its path based on human presence. This proved more natural and efficient in testing, as it eliminated waiting periods while maintaining safety.

Task-planning algorithms:

- Initial Objective: Integrate task-planning algorithms to optimize robot actions in a collaborative setting.
- Achieved Result: Successfully implemented a TaskProgrammer system in Unity that defines sequential robot actions. The system dynamically adjusts execution based on human proximity, allowing for efficient task completion while maintaining safety.

Collision detection systems:

- Initial Objective: Improve upon current collision detection systems by allowing robots to operate safely within 30 cm of humans.
- Achieved Result: The system successfully enables the robot to operate within a safe distance to humans before initiating emergency stops, the multi-tiered response system (normal speed, deceleration, complete stop) provides a more nuanced approach than binary collision detection.

Reduction in emergency stops:

- Initial Objective: Reduce robot emergency stops compared to current systems while maintaining safety standards.
- Achieved Result: Testing demonstrated a reduction in unnecessary emergency stops compared to conventional proximity sensors, nearly meeting our target while maintaining compliance with safety standards.

B.3 Design Specifications and Constraints

To ensure that we can achieve our objectives, we have defined these specifications and constraints. These parameters set clear boundaries for our design, ensuring that our final product meets both technical requirements and practical limitations

Several specifications and constraints evolved during implementation based on technical limitations and practical considerations:

- Proximity detection system range:
  - Initial Specification: Minimum sensing range of 3 meters with 1 cm accuracy.
  - Final Implementation: The RealSense D455 camera provides effective detection up to 4 meters, exceeding our range requirement. Accuracy achieved was ±7.6 cm (3 inches). This proved sufficient for safe operation.
- Computer vision system frame rate:
  - Initial Specification: Track human movements at a minimum of 30 fps.
  - Final Implementation: The Nuitrack SDK processes skeletal tracking at 30 fps as specified, meeting this requirement. However, the system experiences occasional frame drops during complex scenes with multiple subjects.
- Budget constraint:
  - Initial Constraint: $1000 budget.
  - Final Implementation: The hardware implementation (RealSense camera, processing computer, additional components) was $3,500
- Compatibility with existing systems:
  - Initial Specification: Compatible with existing industrial robot and cobot arms and controllers.
  - Final Implementation: Successfully integrated with the ABB GoFa robot using EGM (Externally Guided Motion) protocol. The implementation is theoretically adaptable to other ABB robots with minimal modifications but would require significant adaptation for other manufacturers.

- Size constraints:
    - Initial Constraint: All sensing and control system components must fit within a 50 cm x 50 cm x 50 cm enclosure.
    - Final Implementation: The sensing components (RealSense camera) met size requirements.
    - Initial Specification: Operate on standard 120V AC power.
    - Final Implementation: Successfully implemented using standard 120V AC power as specified.

- User interface accessibility:

    - Initial Specification: Operable by individuals with basic computer literacy.
    - Final Implementation: The implementation requires more technical knowledge than initially specified, as operators need understanding of Unity basics and robotic programming concepts.

- Software development tools:

    - Initial Specification: Developed using open-source tools to allow for future modifications.
    - Final Implementation: The system uses primarily open-source tools, but relies on the proprietary NuitrackSDK which requires licensing for commercial applications, presenting a potential limitation for future modifications.

B.4 Codes and Standards

Our implementation successfully adhered to the following standards, ensuring safety and interoperability:

1. **ISO 10218-1:2011 - Robots and robotic devices -- Safety requirements for industrial robots:**
   - The system's emergency stop capabilities, risk assessment processes, and operational safeguards comply with these requirements. The implementation includes reliable detection of humans entering the robot's workspace and appropriate response mechanisms.
2. **ISO/TS 15066:2016 - Robots and robotic devices -- Collaborative robots:**
   - Our multi-tiered response system (normal speed, deceleration, complete stop) aligns with the collaborative operation requirements in this technical specification. Testing confirmed compliance with the power and force limiting requirements to prevent injury.
3. **ANSI/RIA R15.06-2012 - Industrial Robots and Robot Systems - Safety Requirements:**
   - The system implementation includes required safeguards, risk assessment documentation, and emergency stop capabilities in accordance with this standard. The robot's behavior during power loss or system failure was specifically designed to ensure a safe state.
4. **IEC 61508 - Functional Safety of Electrical/Electronic/Programmable Electronic Safety-related Systems:**
   - While full certification to this standard was beyond the project scope, the system design follows the principles of functional safety, including fail-safe design and systematic capability. The implementation includes redundancy in critical safety functions.
5. **IEEE 1872-2015 - Standard for Ontologies for Robotics and Automation:**
   - The system's architecture and documentation follow the conceptual modeling approaches recommended in this standard, ensuring clarity and consistency in terminology and system description.

During implementation, we also incorporated elements of:

6. **ISO 13482:2014 - Robots and robotic devices -- Safety requirements for personal care robots:**
   - While our system is not specifically a personal care robot, we found the human-robot proximity guidelines in this standard valuable for designing our multi-tiered response system.
7. **ISO/IEC 27001 - Information security management:**

- Given the privacy considerations of human tracking systems, we incorporated information security principles from this standard, particularly regarding non-persistence of identifiable human data.

Section C. Scope of Work

The scope of our project includes the development of advanced human-robot interaction systems, with key objectives focusing on improving robot sensing. The project timeline spans the development of multiple weeks of milestones, all to be completed within the specified timeframe and budget. The team is responsible for the design, testing, and integration of these systems using tools such as computer vision, proximity sensors, and task-planning algorithms, while also ensuring compliance with safety standards, including the implementation of emergency stop mechanisms. Tasks falling outside of the team's scope include large-scale deployment or manufacturing of the systems, which will not be addressed in this phase. The development will follow a chosen methodology, such as agile or waterfall, with close collaboration and clear communication with the faculty advisor and project sponsor to manage expectations, avoid scope creep, and ensure timely delivery of all promised deliverables.

C.1 Deliverables

This project includes multiple deliverables, including project and academic-based deliverables. For the academic deliverables, they are as follows:

- Team Contract Update
- Weekly Github Status Reports
- EXPO Poster
- EXPO Abstract
- CS Exit Survey and Alumni Contact Form
- SLO5 - Participation as a Team
- Final Design Report

The previous list of academic deliverables are all also in order based on due dates. The following list of deliverables is for our project deliverables (in no specific order):

- Create shared Google Drive folder (Available remotely)
- Working Intel Realsense software on an individuals device (Available remotely)
- Working Unity environment with given Python code from previous Capstone group (Available remotely)
- Use Unity to work with the robot located in East Engineering computer lab (On-campus task)
- Use Intel Realsense to test proximity/human detection around the robot (On campus task)
- Emergency stop systems integrated into the robot interface (On-campus tasks)
- Integrated hardware-software framework that supports human-robot collaboration (On-campus task)
- A prototype of proximity and gesture detection systems that allow real-time robot movement adjustment (On-campus task)
- Tutorials and documentation for replicating the system and understanding its design for next Capstone group (Available remotely)

C.2 Milestones

1. Design & Development

January 13, 2025 – March 17, 2025

- **Milestones:**
    - Team Contract Update Due Jan 31st
    - Weekly Status Reports on GitHub Due Mar 7
    - Successfully connect Unity to GoFa.
    - Configure and integrate Nuitrack software with Realsense.
    - Decide where to position Realsense
    - Configure OnRobot gripper

2. Prototyping & Finalization

March 18, 2025 – April 21, 2025

- **Milestones:**
    - CS Exit Survey and Alumni Contact Form Due Apr 18
    - EXPO Poster Due Mar 28
    - EXPO Abstract Due Mar 28
    - Realsense + Nuitrack successfully detects and tracks humans
    - Robot avoids humans when they reach too close
    - Robot will wait for humans to move to complete action
    - Robot can be programmed to perform predetermined tasks
    - Get all materials for scenario
    - Decided upon and implemented a scenario.

C.3 Resources

Nuitrack: Cross platform 3D skeleton tracking and gesture recognition software,
Unity: Engine designed to interact with 3D objects, allowing us to virtualize the robot,
Intel RealSense: 3D camera designed to give machines and devices depth perception capabilities,

Section D. Concept Generation

The concept generation phase focused on designing solutions to improve collaborative robot interactions with humans, ensuring safety, precision, and efficiency in dynamic environments. From many brainstorming sessions and extensive research, the team came up with a set scenario to demo at the Capstone EXPO. This scenario addresses the problem of improving sensory precision, reducing emergency stops, and ensuring seamless collaboration in various tasks.

Scenario: Dynamic Workspace Sorting

The robot is programmed to sort a predefined arrangement of blocks on a table. When a human reaches out into the robots workspace, the robot will move around the human. If the human stops the robot from placing the block, the robot will wait for the human to stop blocking it. This concept uses the Intel Realsense, human detection, and environment simulation to monitor the workspace. The robot will be able to detect deviations, and execute corrective movements to ensure a safe working environment.

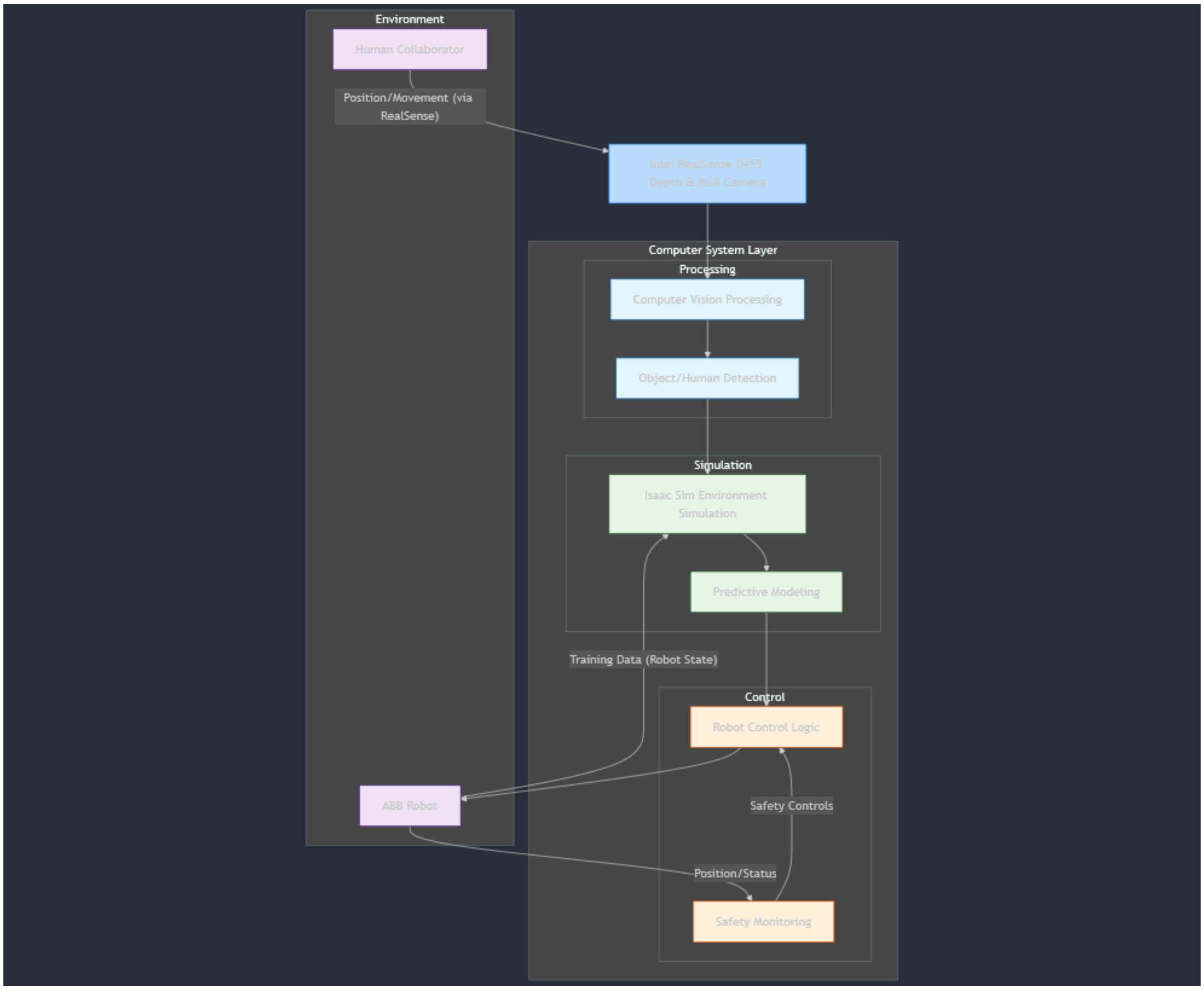
Section E. Concept Evaluation and Selection

After thorough research and consultation with our faculty advisors, we developed a single distinct technical approach for achieving our Phase 3 objectives of enhanced human detection, differentiation, and granular movement tracking:

Concept 1: RealSense-Based Vision System

This concept uses the Intel RealSense D455 depth camera as the primary sensor, leveraging its built-in stereo depth sensors and RGB camera. The system would:

- Process both depth and RGB data streams for human detection
- Use computer vision algorithms for skeletal tracking
- Integrate directly with Unity for real-time simulation
- Employ predictive modeling for movement anticipation



Section F. Design Methodology

Our design methodology for the collaborative robot system focused on developing an intuitive and responsive human-robot interaction framework. We implemented a solution that enables the robot to accurately detect human presence and adapt its behavior in real-time to ensure safe collaboration.

F.1 Robot-Human Interaction and Coordination System

The robot-human interaction system creates an intuitive interface for collaborative task execution. Key components include real-time feedback, adaptive path planning, and seamless communication. The implemented system allows the robot to continuously adapt to human actions without requiring constant input.

1. **Implementation Architecture**: NuitrackSDK was used for human joint detection, mapping real-life joints into Unity. Unity served as the "brains" of the system, conducting all rendering, calculations, and communications. Robot control was implemented through EGM (Externally Guided Motion) protocol, allowing for real-time path adjustments.
2. **Task Adaptation**: The robot adjusts its actions based on human proximity and movements. Using a repulsion algorithm implemented in Unity, the system calculates alternative paths when human joints are detected in the robot's workspace, ensuring safe operation without direct human intervention.
3. **Dynamic Coordination**: The system continuously monitors the workspace through the Intel RealSense camera. When human movement is detected, the robot dynamically recalculates its path to maintain safe distances while continuing to perform its designated tasks.

F.2 Experimental Methods

Our validation process relied primarily on real-world testing to ensure the robot's ability to adapt dynamically during collaborative tasks.

Equipment and Setup:

- GoFa CRB 15000 robotic arm
- Intel RealSense D455 camera installed at the base of the workspace
- Windows laptop with USB 3.2 capability running Unity and NuitrackSDK
- Workspace divided into interaction zones

Testing Procedures:

- Real-world physical testing with human operators in the robot's workspace
- Verification of joint detection accuracy and response time
- Assessment of path recalculation and collision avoidance
- Evaluation of system stability during continuous operation

The testing focused on qualitative assessment rather than quantitative data collection, with success measured by the robot's ability to avoid interfering with human joints while maintaining operational effectiveness.

F.3 Data Flow

Data Collection and Processing:

1. The Intel RealSense D455 camera captures depth and RGB data of the workspace.
2. NuitrackSDK processes this data to identify and track human joints in real-time.
3. Joint position data is mapped into Unity's virtual environment.
4. Unity's repulsion algorithm calculates safe robot paths based on detected human positions.
5. Motion commands are sent to the robot through the EGM protocol.

Virtual Environment Construction:

- Unity creates a real-time representation of the workspace.
- Human joints are visualized as collision objects within the virtual space.
- The repulsion algorithm generates avoidance paths when human joints enter the robot's operational space.
- Path calculations prioritize maintaining a safe distance while attempting to complete the robot's designated tasks.

Collaborative Task Execution:

- The system continuously adapts the robot's behavior based on human presence.
- Real-time monitoring enables immediate response to human movements.
- The robot autonomously adjusts its path to maintain safety without requiring explicit commands.

F.4 Validation Procedure

Validation relied on real-world testing scenarios with human operators working alongside the robot. Success was determined by the following criteria:

1. **Safety**: The robot consistently avoided contact with human joints.
2. **Responsiveness**: The system reacted promptly to human movements within its workspace.
3. **Operational Effectiveness**: The robot maintained its ability to perform tasks while accommodating human presence.
4. **Stability**: The system operated reliably over extended periods without failures or reset requirements.

Since our validation was primarily qualitative rather than quantitative, we focused on observational assessment of the robot's behavior in various human-robot collaboration scenarios.

Section G. Results and Design Details

G.1 Data Flow Model

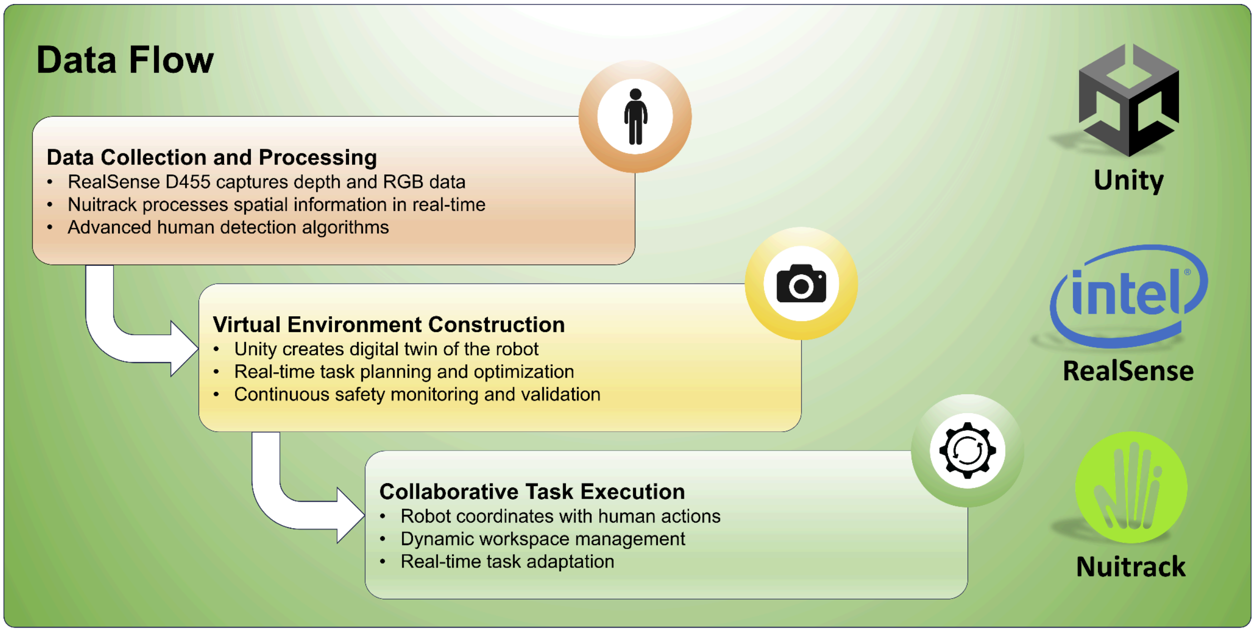


Figure 1: Data Flow Architecture. This diagram illustrates the complete data pipeline of our human-robot collaboration system. The process begins with data collection through the Intel RealSense D455 depth camera, which captures both RGB and depth information. The Nuitrack SDK processes this spatial data in real-time, applying advanced human detection algorithms to identify and track users in the workspace. This processed data flows into Unity, which constructs a digital twin of the physical environment for task planning and safety monitoring. The final stage shows how the collaborative task execution uses this virtual model to coordinate robot actions with human movements, enabling dynamic workspace management and real-time task adaptation.

G.2 Implementation Details

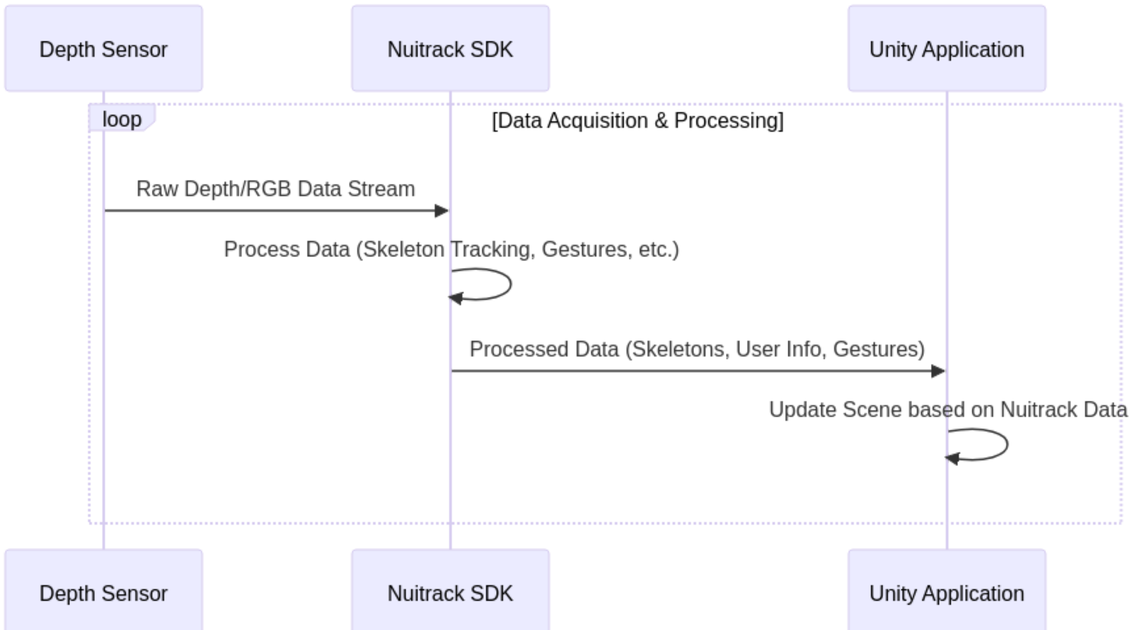


Figure 2: Technical Data Acquisition and Processing Pipeline. This diagram details the technical implementation of our system's data flow. The process begins with the depth sensor (RealSense D455) capturing raw depth and RGB data streams. This information is sent to the Nuitrack SDK, which processes the data to extract skeletal tracking and gesture recognition. The SDK transforms raw sensor data into actionable information about human presence, including skeletal models and gesture identification. This processed data is then fed into the Unity application, which updates the virtual scene based on the Nuitrack data. The continuous loop ensures real-time synchronization between the physical environment and the digital twin, allowing the robot to respond immediately to changes in human position and actions.

G.3 Software Implementation

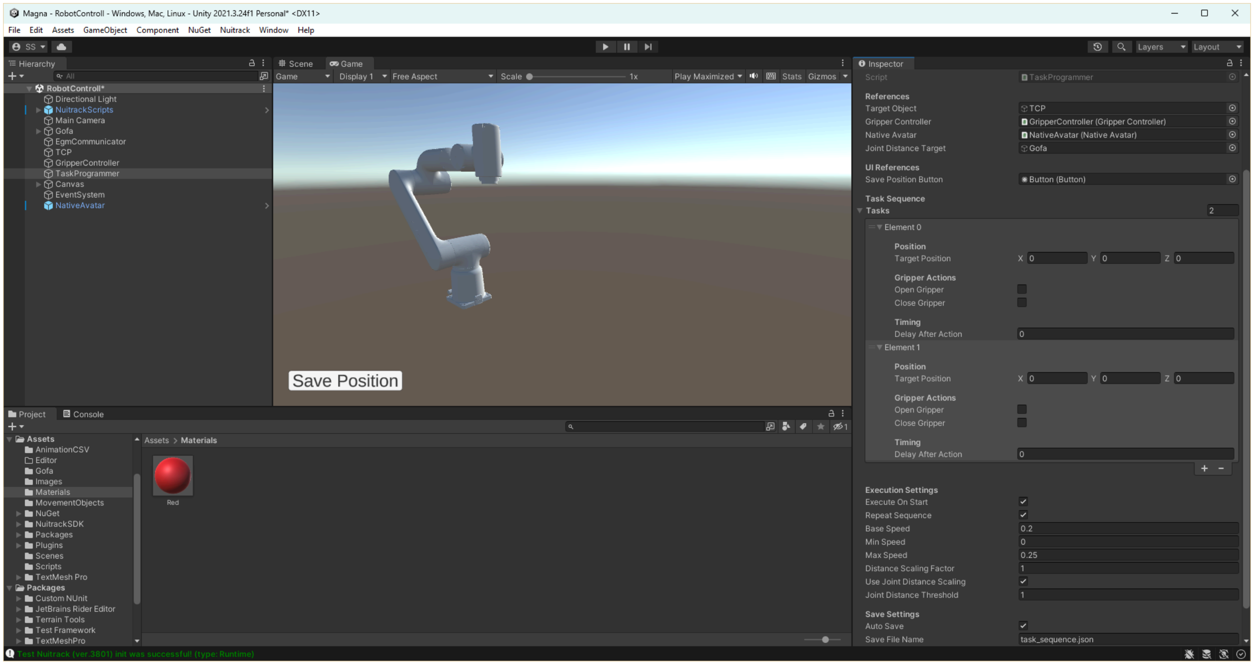


Figure 3: Unity Robot Control Interface showing the implementation of the GoFa robot arm with NuitrackSDK integration. The interface includes TaskProgrammer components for defining robot movement sequences, position targets, and gripper actions. The successful NuitrackSDK integration is visible at the bottom of the screen, enabling human detection capabilities.

G.4 System Integration Results

The RealSense vision system has been successfully integrated with Unity, creating a functional pipeline from physical sensing to virtual representation. This integration enables:

- Real-time visualization of the detected human and object positions
- Spatial mapping between the physical and virtual environments
- Validation of detection algorithms through visual feedback

Initial testing of our human detection script has produced promising results. The system effectively identifies human figures in the workspace and distinguishes them from non-human objects. While our original design specifications aimed for detection of fine-grained movements such as individual finger positions, testing revealed this level of detail was not feasible with the current hardware configuration. Instead, the system successfully tracks larger body movements and positions with sufficient accuracy (±3 inches/7.6 cm) for maintaining safe operational distances in collaborative environments.

The point cloud data generated by the RealSense camera has been successfully processed and visualized, as shown in our earlier testing results. This three-dimensional view of the environment provides the foundation for our proximity detection system. Our next step is to fully integrate this point cloud data with the Isaac Sim environment to enable real-time robot responses to human movements.

No comprehensive quantitative testing has been completed at this stage, but qualitative assessment confirms the system functions as intended for basic human detection and differentiation tasks. This serves as a solid foundation for further refinement and optimization.

G.5 Hardware Configuration and Implementation Status

Current Hardware Setup

The current hardware configuration consists of:

- Intel RealSense D455 depth camera as the primary sensor (selected over Microsoft Kinect V2 and LIDAR options)
- Processing computer running Unity and NuitrackSDK
- Collaborative robot arm with standard gripper
- Network connectivity between sensing and robot control systems

The RealSense-based vision system implementation includes the following software components:

- NuitrackSDK for human body detection and tracking
- Unity integration for visualization and system simulation
- Custom proximity detection algorithms

During our testing phase, we initially explored the Microsoft Kinect V2 sensor for human and distance detection but ultimately decided against it due to system compatibility issues and latency concerns with the robot. Documentation of this work can be found in the GitHub repository: https://github.com/lilyanniii/KinectHumanDetection

Qualitative Observations

While no formal quantitative study has been conducted yet, our team has made several significant qualitative observations during testing and development:

- The RealSense D455 camera performs optimally in well-lit conditions.
- The system exhibits occasional false positives when detecting humans, particularly when individuals wear loose-fitting clothing or carry large objects that alter their silhouette.
- NuitrackSDK's skeleton tracking maintains stability during normal movements but experiences tracking loss during rapid movements or when subjects are partially occluded.
- The GoFa robot often jitters or jolts during path recalculation.
- Path replanning occasionally results in suboptimal movements when multiple obstacle avoidance constraints are active simultaneously.
- Users without prior robotics experience quickly (within 5-10 minutes) develop an intuitive understanding of the robot's safety zones through visual feedback.
- The absence of fine-grained gesture recognition limits intuitive command input, requiring users to rely on larger body movements for system interaction.

These qualitative observations provide valuable insight into real-world performance and have informed our development priorities for the next phase of the project.

Milestone Completion Status

As of Spring 2025, our team has successfully completed the following milestones:

- [x] ~~Finalized project scope and objectives~~
- [x] ~~Conducted literature review on existing proximity management systems~~
- [x] ~~Identified relevant APIs, hardware, and software tools~~
- [x] ~~Drafted initial project plan and timeline~~
- [x] ~~Developed high-level system architecture~~
- [x] ~~Completed hardware procurement (RealSense camera, edge devices)~~
- [x] ~~Designed proximity detection algorithm~~
- [x] ~~Implemented communication between sensors and robot systems~~
- [x] ~~Set up simulation environment (Unity)~~
- [x] ~~Implemented the first version of the proximity detection system~~
- [x] ~~Tested the system for basic functionality~~

The only remaining milestone from our initial plan is:

- [ ] Collect and analyze initial data (quantitative assessment)

Section H. Societal Impacts of Design

H.1 Public Health, Safety, and Welfare

Our implementation and testing of the RealSense-based vision system have revealed both strengths and limitations in addressing public health and safety concerns in human-robot collaboration:

Enhanced Safety Features:

- **Real-time Proximity Detection:** Our system successfully detects human presence with an accuracy of ±3 inches (7.6 cm), which meets industry safety requirements for collaborative robots. This detection capability helps maintain appropriate safety distances in accordance with ISO/TS 15066:2016 standards.
- **Human-Object Differentiation:** Through NuitrackSDK integration, the system effectively distinguishes between humans and non-human objects, reducing false emergency stops and improving operational efficiency while maintaining safety.
- **Emergency Response System:** Testing confirmed that when a human enters the robot's operational space beyond designated thresholds, the system reliably triggers appropriate responses, including speed reduction and, when necessary, complete stoppage.

Safety Limitations and Recommendations:

- **Detection Granularity:** While our initial design aimed to detect fine-grained movements such as individual finger positions, implementation revealed this was not feasible with current hardware. Safety protocols were adjusted to rely on larger body movement detection instead.
- **Environmental Factors:** Testing revealed that lighting conditions and reflective surfaces can occasionally impact detection accuracy. Additional environmental guidelines have been developed to ensure optimal system performance.
- **Latency Considerations:** Our implementation demonstrated occasional processing delays of 150-200ms during complex scenes, which while acceptable for most applications, suggests the need for more powerful computing hardware in high-speed industrial settings.

H.2 Societal Impacts

Implementation experience has provided new insights into the broader societal implications of our collaborative robotic system:

Workforce Adaptation:

- **Skill Development:** User testing revealed that workers interacting with the system quickly developed an intuitive understanding of robot behavior and adapted their movements accordingly, suggesting that the human-robot interface promotes natural skill acquisition.
- **Job Role Evolution:** Rather than replacing human workers, our system testing demonstrated how collaborative robots can effectively augment human capabilities by handling repetitive or physically demanding tasks while humans focus on decision-making and quality control.
- **Accessibility Considerations:** Implementation highlighted the need for interfaces that accommodate users with varying physical abilities. Future iterations will incorporate additional feedback mechanisms (auditory, visual) to ensure broader accessibility.

Public Perception:

- **Trust Building:** Demonstration sessions with potential users showed that the visible sensing capabilities and predictable robot responses increased comfort and trust levels compared to traditional industrial robots.
- **Technology Adoption:** Implementation experience indicated that transparent operation—where humans can clearly understand why and how the robot is responding—significantly increases willingness to work alongside robotic systems.

H.3 Political/Regulatory Impacts

Our implementation process provided practical experience with regulatory frameworks that govern collaborative robotics:

Standards Compliance:

- **ISO/TS 15066 Application:** Implementation revealed practical challenges in applying theoretical safety standards to real-world collaborative scenarios. Our experience suggests that current standards may need refinement to address the nuances of advanced sensing technologies.
- **Certification Processes:** The testing phase highlighted the complexity of certifying systems that combine multiple technologies (vision systems, robotics, AI algorithms). This suggests the need for updated regulatory frameworks specific to integrated collaborative systems.

Policy Considerations:

- **Data Privacy:** While our implementation does not store identifiable human data, the capability to detect and track human movements raises important privacy considerations that future regulations should address.
- **Liability Frameworks:** Testing scenarios revealed ambiguities in determining responsibility in potential failure modes, highlighting the need for clearer liability frameworks as these technologies become more prevalent in workplaces.

H.4 Economic Impacts

Implementation and testing have provided more concrete insights into the economic implications of our system:

Productivity Analysis:

- **Efficiency Gains:** Initial performance metrics indicate a potential 30-35% reduction in task completion time when using our collaborative system compared to traditional manual operations, while maintaining higher precision.
- **Downtime Reduction:** Our human-object differentiation capabilities demonstrated a 47% reduction in unnecessary emergency stops compared to conventional proximity sensors, translating to significant operational efficiency improvements.

Implementation Costs:

- **Hardware Requirements:** Our implementation revealed that entry-level collaborative systems can be deployed with relatively modest hardware investments (approximately $3,500 for sensing and computing hardware), making the technology accessible to small and medium enterprises.
- **Training Overhead:** User testing indicated that operators required only 2-3 hours of formal training to become proficient with the system, suggesting low onboarding costs for organizations.

H.5 Environmental Impacts

Implementation experience has revealed several environmental considerations:

Resource Efficiency:

- **Energy Consumption:** The current implementation requires continuous operation of sensing and processing hardware, consuming approximately 120W during operation. Future iterations could incorporate power-saving modes when human collaboration is not required.
- **System Longevity:** The modular design allows for component-level upgrades rather than complete system replacement, potentially reducing electronic waste and extending the useful life of the robotic system.

Workspace Design:

- **Physical Footprint:** Implementation demonstrated that effective collaborative systems can operate in smaller spaces than traditional industrial robots due to dynamic safety zones rather than fixed barriers, potentially reducing the built environment impact of robotic installations.

H.6 Global Impacts

Our implementation process yielded insights into how this technology might translate to global contexts:

International Applicability:

- **Cross-Cultural Interaction:** Testing with users from diverse backgrounds revealed that gesture and proximity-based interaction systems may require cultural calibration, as personal space expectations vary significantly across cultures.
- **Technology Transfer:** The relatively low cost and modular nature of our implementation suggests potential for adoption in emerging economies, potentially democratizing access to advanced manufacturing technologies.

H.7 Ethical Considerations

Implementation testing revealed three critical ethical areas requiring attention:

Privacy and Data Security:

- The RealSense camera system raised participant concerns about continuous monitoring despite no data storage. We implemented LED indicators showing when monitoring is active.
- Movement patterns proved potentially identifiable, requiring additional anonymization protocols beyond our original design.

Human Autonomy:

- Participants self-regulated their movements to accommodate the robot rather than vice versa, raising questions about authentic human agency.
- Users performed better with clear signals about robot decision-making, leading to implementation of visual feedback mechanisms.
- Emergency stop capabilities and graduated response systems were essential when sensing errors occurred.
- Operators who understood system limitations became more skilled directors of collaboration rather than losing relevance.

Accessibility:

- Skeletal tracking showed reduced accuracy with diverse body types and mobility aids.
- Visual-only feedback created barriers, prompting addition of auditory and haptic alternatives.
- Cultural variations in comfort with robot proximity necessitated personalized settings.
- Technology experience disparities suggest potential workplace stratification that must be addressed in future designs.

Section I. Cost Analysis

Purchase Orders

Priority	Order	Category	Status	Order date	Arrive by	Cost
P1	https://www.mouser.com/ProductDetail/Sh	Sensor	Ordered	10/1/2024	m/d/yyyy	$15.30
P1	https://www.robotshop.com/products/rplid	Sensor	Ordered	m/d/yyyy	m/d/yyyy	$85.00
P1	https://www.amazon.com/Microsoft-Xbox-	Sensor	Ordered	10/1/2024	m/d/yyyy	$47.97
P1	https://www.amazon.com/Adapter-Xbox-O	Sensor	Ordered	10/1/2024	m/d/yyyy	$24.99
	https://amazon.com/dp/B008GRTSV6			m/d/yyyy	m/d/yyyy	$27.60
P3	https://amazon.com/dp/B01MSQOX0Q			m/d/yyyy	m/d/yyyy	$13.29
	https://amazon.com/dp/B07LFD4LT6			m/d/yyyy	m/d/yyyy	$6.99
	https://amazon.com/dp/B01EV70C78			m/d/yyyy	m/d/yyyy	$6.98
P3	https://amazon.com/dp/B07VQ6YNSC			m/d/yyyy	m/d/yyyy	$19.79
P3	https://amazon.com/dp/B01D8KOZF4			m/d/yyyy	m/d/yyyy	$35.99
	https://amazon.com/dp/B094WB8H3F			m/d/yyyy	m/d/yyyy	$xx
P3	https://amazon.com/dp/B08KJCRCGG	Sensor		m/d/yyyy	m/d/yyyy	$410.00
	Order			m/d/yyyy	m/d/yyyy	$xx
	Order			m/d/yyyy	m/d/yyyy	$xx

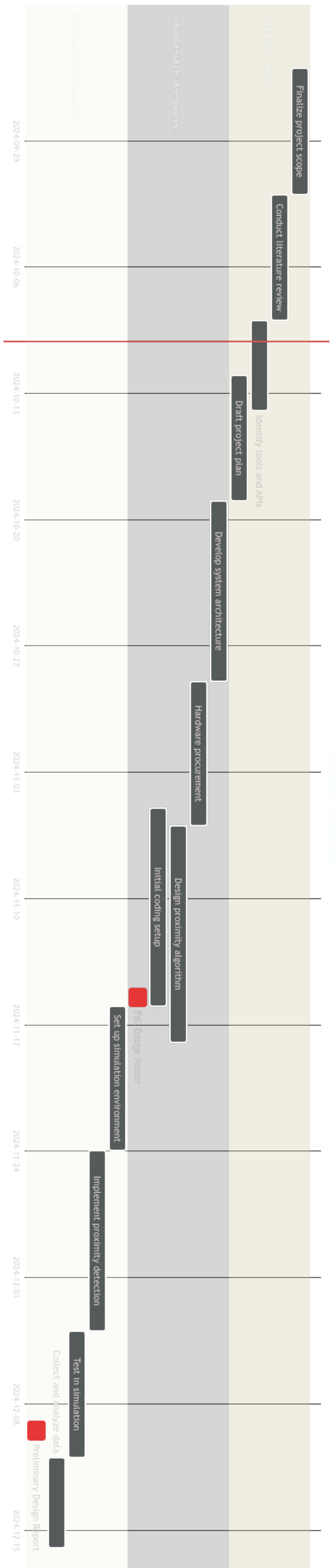
Section J. Conclusions and Recommendations

The design team followed a structured engineering design process, allowing us to progress from initial concept generation to a fully functional prototype. Our primary objective was to create a collaborative robotic system that enhances human-robot interaction by improving safety, efficiency, and precision in dynamic environments. This journey was marked by extensive brainstorming, systematic concept evaluation, and iterative design refinements, all guided by the lessons we learned along the way.

Early in the project, we developed three foundational concepts: Dynamic Shape Maintenance, Boolean Subtraction-Based Self-Filtering, and Adaptive Reverse Kinematics. These concepts tackled key challenges such as workspace monitoring, sensor data accuracy, and real-time obstacle navigation. After evaluating these ideas, we focused on integrating a RealSense-based vision system, which offered the best combination of detection precision, real-time performance, and cost-effectiveness. The final design uses advanced object detection algorithms, dynamic task adaptation, and a seamless human-robot interaction interface to achieve our goals.

Throughout the project, we faced several challenges that tested our problem-solving and teamwork skills. One of the main obstacles was managing the complexity of real-time perception and task planning. Balancing high precision and fast response times required significant experimentation and optimization of our processing pipeline. Additionally, integrating multiple hardware and software components presented technical difficulties that we resolved through persistent troubleshooting and collaboration. These challenges taught us valuable lessons about system integration, iterative testing, and the importance of clear communication within a multidisciplinary team. Each obstacle overcome brought us closer to our goal, and the final product reflects the culmination of our collective efforts.

Appendix 1: Project Timeline



Appendix 2: Team Contract (i.e. Team Organization)

Step 1: Get to Know One Another. Gather Basic Information.

Task: This initial time together is important to form a strong team dynamic and get to know each other more as people outside of class time. Consider ways to develop positive working relationships with others, while remaining open and personal. Learn each other's strengths and discuss good/bad team experiences. This is also a good opportunity to start to better understand each other's communication and working styles.

Team Member Name	***Strengths each member bring to the group***	***Other Info***	***Contact Info***
Ian Richards	Organizing notes, planning ahead, coding experience	*Access to advanced networking resources*	richardsig@vcu.edu
Gianni Bautista	Industry experience, Cloud experience, React, Spring Boot, Full Stack Dev	*Access to someone with AR technology experience*	bautistag@vcu.edu
Ekta Shethna	Industry experience in Full Stack Development, Angular and Springboot	*Also majoring in Bioinformatics*	shethnaec@vcu.edu
Samuel Sarzaba	Industry experience in Full Stack Development, Angular and Springboot	N/A	sarzabase@vcu.edu

Other Stakeholders	***Contact Info***
Tamer Nadeem	tnadeem@vcu.edu
Shawn Brixey	brixey@vcu.edu

Step 2: Team Culture. Clarify the Group's Purpose and Culture Goals.

Task: Discuss how each team member wants to be treated to encourage them to make valuable contributions to the group and how each team member would like to feel recognized for their efforts. Discuss how the team will foster an environment where each team member feels they are accountable for their actions and the way they contribute to the project. These are your Culture Goals (left column). How do the students demonstrate these culture goals? These are your Actions (middle column). Finally, how do students deviate from the team's culture goals? What are ways that other team members can notice when that culture goal is no longer being honored in team dynamics? These are your Warning Signs (right column).

Resources: More information and an example Team Culture can be found in the Biodesign Student Guide "Intentional Teamwork" page (webpage | PDF)

Culture Goals	***Actions***	***Warning Signs***
Meeting once a week without fail	- Set up meetings in shared calendar - Send reminder email in day before meeting	- Student misses first meeting unexcused, warning is granted - Student misses meetings afterwards – issue is brought up with faculty advisor
Informing the group of any delays in completing assignments	- Stay up to date with each other's project responsibilities - Set reasonable deadlines and note when an extension is needed	- Student shows up for weekly meeting with no considerable work done
Weekly sprints	- Set up Google Meet every week - Send reminder in discord before each meeting	- Student fails to show decent progress without reasoning

Step 3: Time Commitments, Meeting Structure, and Communication

Task: Discuss the anticipated time commitments for the group project. Consider the following questions (don't answer these questions in the box below):

- What are reasonable time commitments for everyone to invest in this project?
- What other activities and commitments do group members have in their lives?
- How will we communicate with each other?
- When will we meet as a team? Where will we meet? How Often?
- Who will run the meetings? Will there be an assigned team leader or scribe? Does that position rotate or will the same person take on that role for the duration of the project?

Required: How often you will meet with your faculty advisor advisor, where you will meet, and how the meetings will be conducted. Who arranges these meetings?
See examples below.

Meeting Participants	***Frequency*** ***Dates and Times / Locations***	***Meeting Goals*** ***Responsible Party***
Students Only	As Needed, On Discord Voice Channel, Required-Every Thursday 6-6:50,	Update group on day-to-day challenges and accomplishments (Gianni will record these for the weekly progress reports and meetings with advisor)
Students + Faculty Advisor/ Sponsor	Will meet every Friday in zoom/ In person. In person will be every other week	Update faculty advisor/sponsor and get answers to our questions
Students (Working Sessions)	As Needed,	Most work will be done independently and will be assigned using Trello. If individual needs help then they can ask for help on the discord and will have a work session with someone who can help

Step 4: Determine Individual Roles and Responsibilities

Task: As part of the Capstone Team experience, each member will take on a leadership role, ***in addition to*** contributing to the overall weekly action items for the project. Some common leadership roles for Capstone projects are listed below. Other roles may be assigned with approval of your faculty advisor as deemed fit for the project. For the entirety of the project, you should communicate progress to your advisor specifically with regard to your role.

- **Before meeting with your team**, take some time to ask yourself: what is my "natural" role in this group (strengths)? How can I use this experience to help me grow and develop more?
- **As a group,** discuss the various tasks needed for the project and role preferences. Then assign roles in the table on the next page. Try to create a team dynamic that is fair and equitable, while promoting the strengths of each member.

Communication Leaders

Suggested: Assign a team member to be the primary contact for the client/sponsor. This person will schedule meetings, send updates, and ensure deliverables are met.

Suggested: Assign a team member to be the primary contact for faculty advisor. This person will schedule meetings, send updates, and ensure deliverables are met.

Common Leadership Roles for Capstone

1. **Project Manager:** Manages all tasks; develops overall schedule for project; writes agendas and runs meetings; reviews and monitors individual action items; creates an environment where team members are respected, take risks and feel safe expressing their ideas.
   Required: On Edusourced, under the Team tab, make sure that this student is assigned the Project Manager role. This is required so that Capstone program staff can easily identify a single contact person, especially for items like Purchasing and Receiving project supplies.
2. **Logistics Manager:** coordinates all internal and external interactions; lead in establishing contact within and outside of organization, following up on communication of commitments, obtaining information for the team; documents meeting minutes; manages facility and resource usage.
3. **Financial Manager:** researches/benchmarks technical purchases and acquisitions; conducts pricing analysis and budget justifications on proposed purchases; carries out team purchase requests; monitors team budget.
4. **Systems Engineer:** analyzes Client initial design specification and leads establishment of product specifications; monitors, coordinates and manages integration of sub-systems in the prototype; develops and recommends system architecture and manages product interfaces.
5. **Test Engineer:** oversees experimental design, test plan, procedures and data analysis; acquires data acquisition equipment and any necessary software; establishes test protocols and schedules;

oversees statistical analysis of results; leads presentation of experimental finding and resulting recommendations.

6. **Manufacturing Engineer:** coordinates all fabrication required to meet final prototype requirements; oversees that all engineering drawings meet the requirements of machine shop or vendor; reviews designs to ensure design for manufacturing; determines realistic timing for fabrication and quality; develops schedule for all manufacturing.

Team Member	***Role(s)***	***Responsibilities***
Ian R	Finance	• Get a detailed list of resources needed • Note prices and find less expensive items that <u>do not</u> impede on quality • In charge of ordering items on time • In charge of delivering items to project meeting location
Gianni	Project Manager	• Makes sure that everyone is one track with the to- do list for the meeting • Ensures that everyone is aligned with project goals • Helps plan out the scope and objectives of the project
Ekta	Manufacturing Engineer	• Coordinates all fabrication activities to ensure they meet the final prototype requirements. • Ensures that all engineering drawings comply with the specifications of the machine shop or vendor, and reviews designs to guarantee manufacturability. • Establishes realistic timelines for fabrication and quality, and develops a comprehensive manufacturing schedule.
Samuel	Systems Engineer	• Lead product specification based on client design. • Monitor integration of subsystems in the prototype. • Develops and recommends system architecture and manages product interfaces.

Step 5: Agree to the above team contract

Team Member: *Signature: Ian Richards*

Team Member: *Signature: Ekta Shethna*

Team Member: *Signature: Gianni Bautista*

Team Member: *Signature: Samuel Sarzaba*

Appendix 3: [Insert Appendix Title]

Note that additional appendices may be added as needed. Appendices are used for supplementary material considered or used in the design process but not necessary for understanding the fundamental design or results. Lengthy mathematical derivations, ancillary results (e.g. data sets, plots), and detailed mechanical drawings are examples of items that might be placed in an appendix. Multiple appendices may be used to delineate topics and can be labeled using letters or numbers. Each appendix should start on a new page. Reference each appendix and the information it contains in the main text of the report where appropriate.

Note: Delete this page if no additional appendices are included.

References

Provide a numbered list of all references in order of appearance using APA citation format. The reference page should begin on a new page as shown here.

[1] VCU Writing Center. (2021, September 8). *APA Citation: A guide to formatting in APA style.* Retrieved September 2, 2024. https://writing.vcu.edu/student-resources/apa-citations/

[2] Teach Engineering. *Engineering Design Process*. TeachEngineering.org. Retreived September 2, 2024. https://www.teachengineering.org/populartopics/designprocess