Video Game Environment for Arm Assessment through a Robotic Exoskeleton

ENGINEERING RELASE REVIEW

Team MAD SABINO

Miguel Villanueva (CS)
Alex Peña (CS)
Dawson Hill (CS)

BACKGROUND

- The University of Idaho Assistive Robotics Lab
- The BLUE SABINO

BLUE **BiLateral Upper-limb Exoskeleton for**

SABINO **Simultaneous Assessment of BIomechanics and Neuromuscular Output**

BLUE SABINO DETAILS

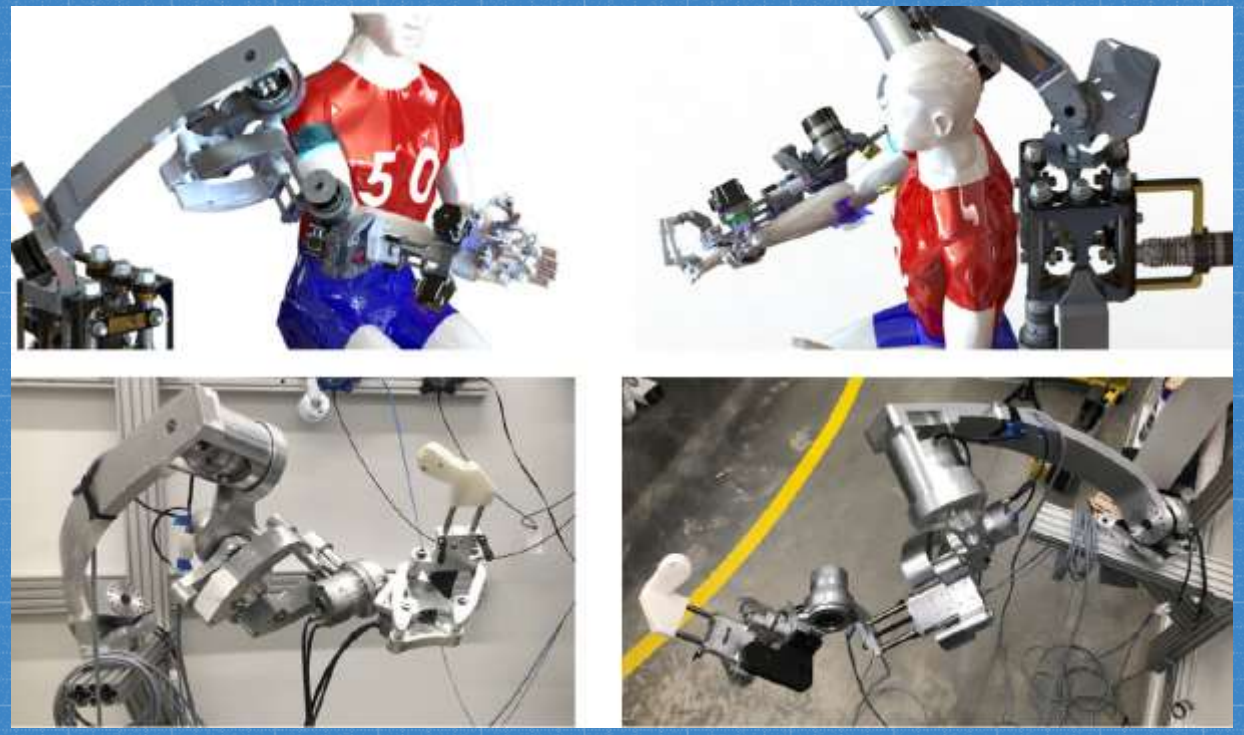
- Being Built in a Series of Phases
- 30 Joint Bilateral Arm and Hand Instrument
- 15 Joints/Motors for Each Arm

<u>Current Phases:</u>

2 joints

5 joints

7 joints

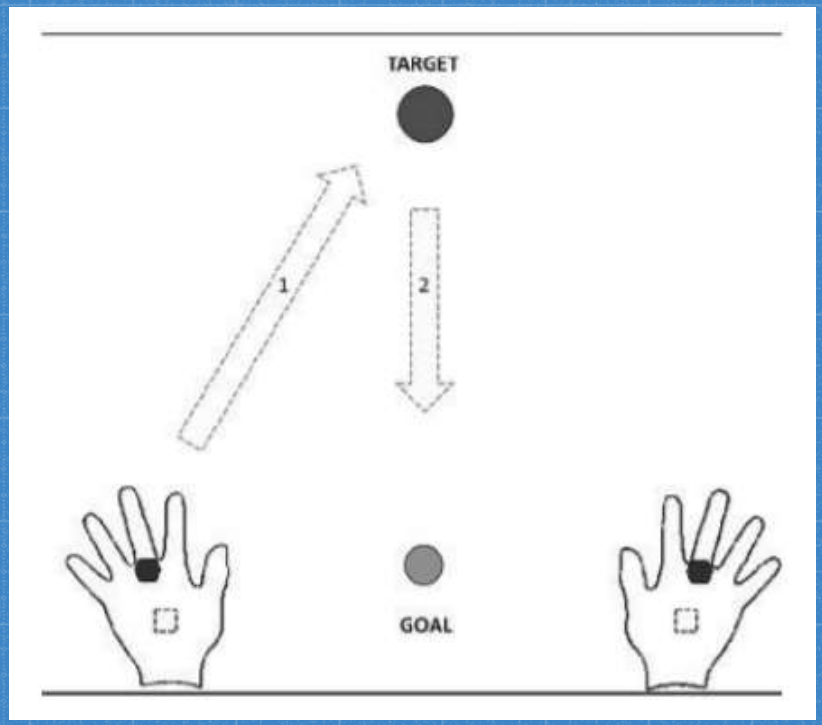


WHAT THE BLUE SABINO DOES

- Arm and Hand Movement Assessment
- Patients with Neurological Impairment (e.g., Stroke)
- ROM = Range of Motion
- COM = Control of Motion
- ROF = Range of Force
- COF = Control of Force

Point-to-point Reach Assessment/Task

- How Steadily (Control of Motion)
- How Fast (Force)

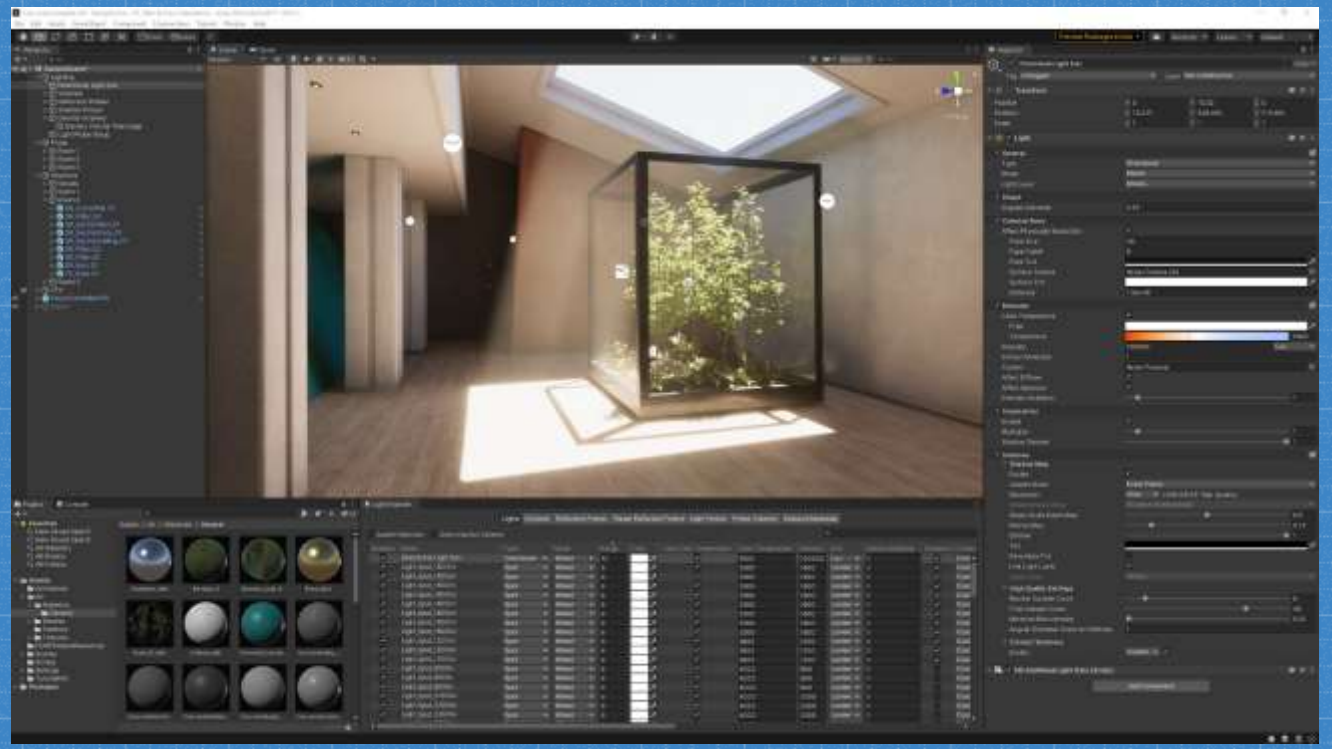


VALUE PROPOSITION

- Important to Provide Clear and Repeatable Tasks
- Auditory Cues and Visual Demonstrations Not Sufficient
- Need a Task Environment

SINGLE OBJECTIVE

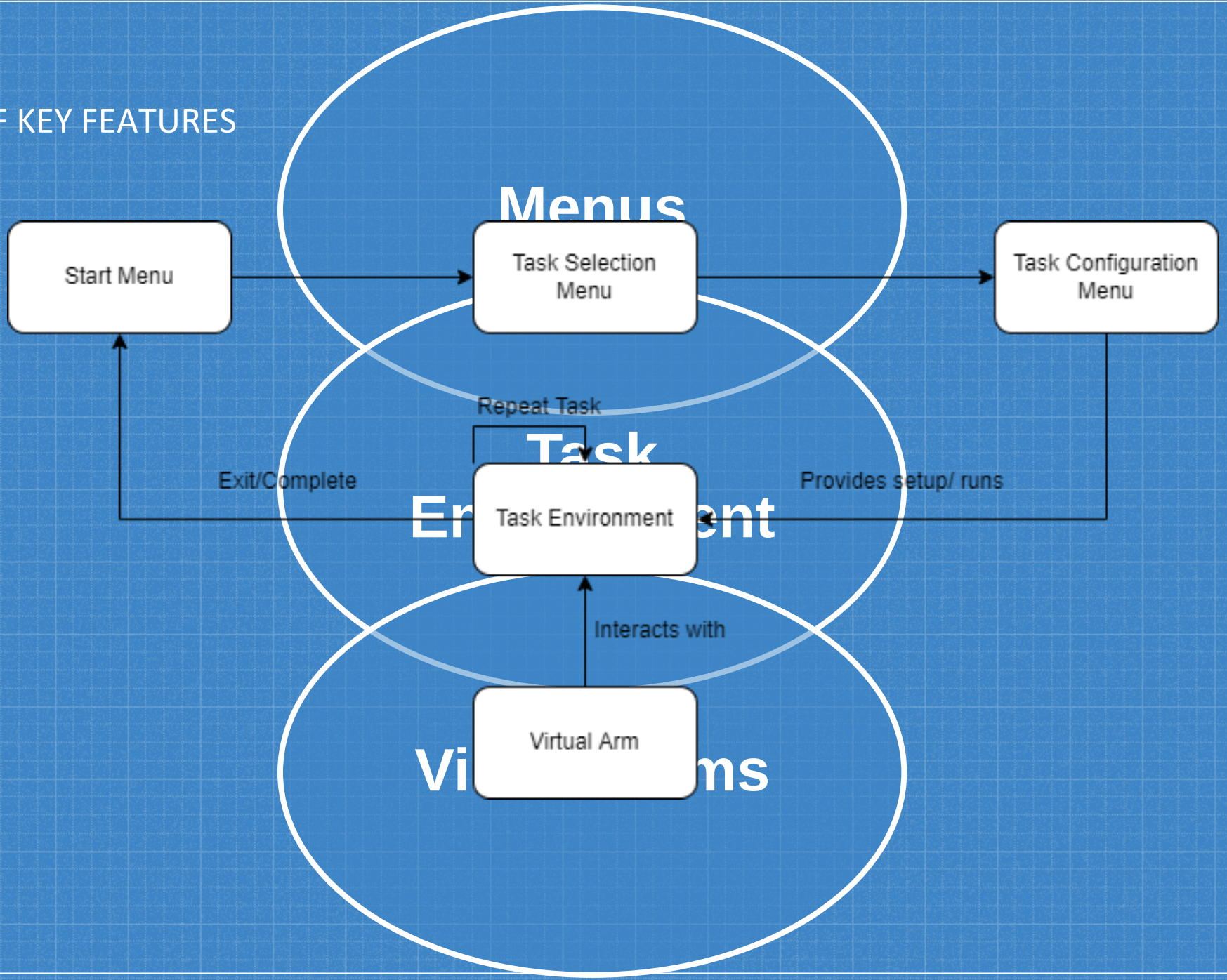
- Build a 3D Virtual Gaming Environment for Arm Assessment
  - Configurable
  - Consistent and Repeatable Task Guidance and Feedback



REQUIREMENTS

- Consistency and Repeatability
- Easily and Highly Configurable Task Settings
- Clear Communication of On-Screen Objects
- Point-to-Point Reach Task
- Virtual Arm Can Modularly Use Multiple Joints

OVERVIEW OF KEY FEATURES



FEATURE – MENUS - FRONTEND

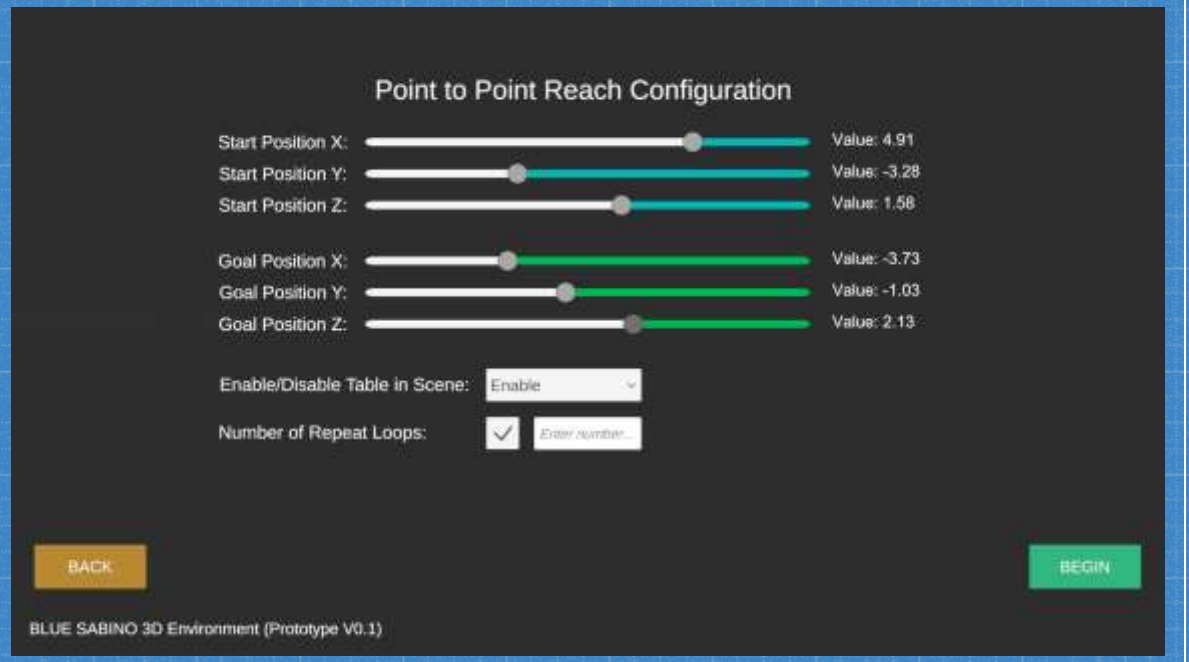
- Application menus must be intuitive and polished
  - *The basic flow and functionality of the menus is implemented; further thought will be put into aesthetics. UI elements have distinct feedback features.*
  - *Input data is reported clearly to the user.*

- Configuration menus need to change parameters within the environment
  - *Task system is in place, now we just need to pass the values to the data container.*

- Configuration menus will need to be customized to each task
  - *Working on finishing Point to Point reach task, this will serve as a foundation to build off for other tasks.*
  - *Menus are implemented in a modular way, so further development work can be streamlined. Unity has good tools for this.*

FEATURE – MENUS - FRONTEND

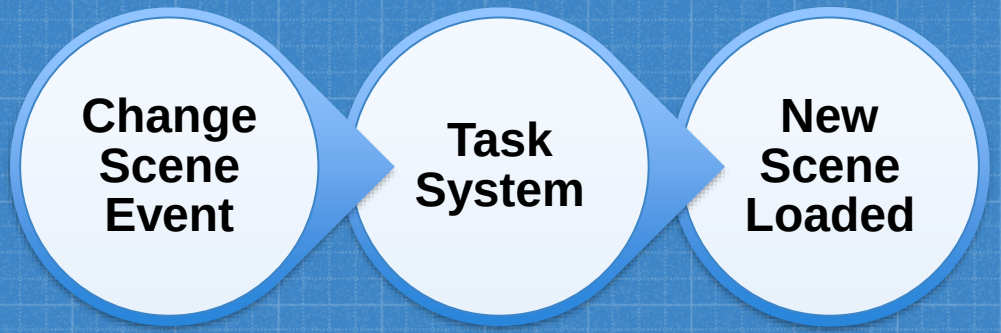
- To do:
  - *Implement in-scene pause menu, ability to return to main menu*
  - *Implement graphics settings, resolution etc.*
  - *Continue polishing aesthetics*
  - *Use data containers to pass parameters through to scene*
  - *Implement text input for slider values*
  - *Map slider values to position values in the environment*
  - *Finalize scene parameters with design team/Dr. Perry*



FEATURE – MENUS - BACKEND

Scene switching

- Task System listens

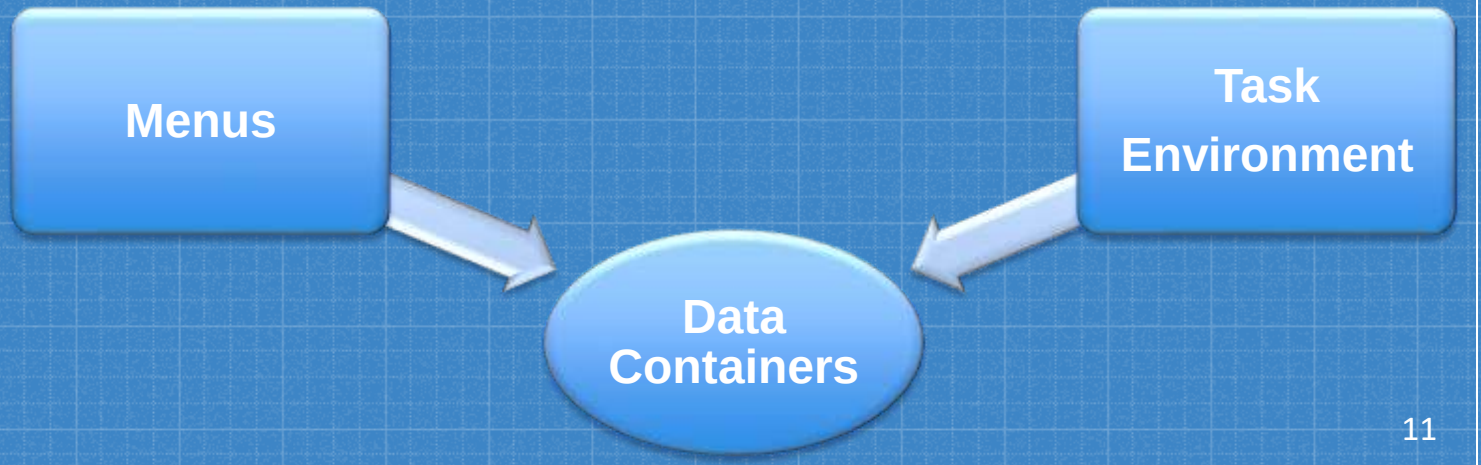


Task selection/configuration

- Data containers
- UI can connect to them

Data containers

- Config
  - Object positions
  - # of repeats
  - Camera position
  - Etc.



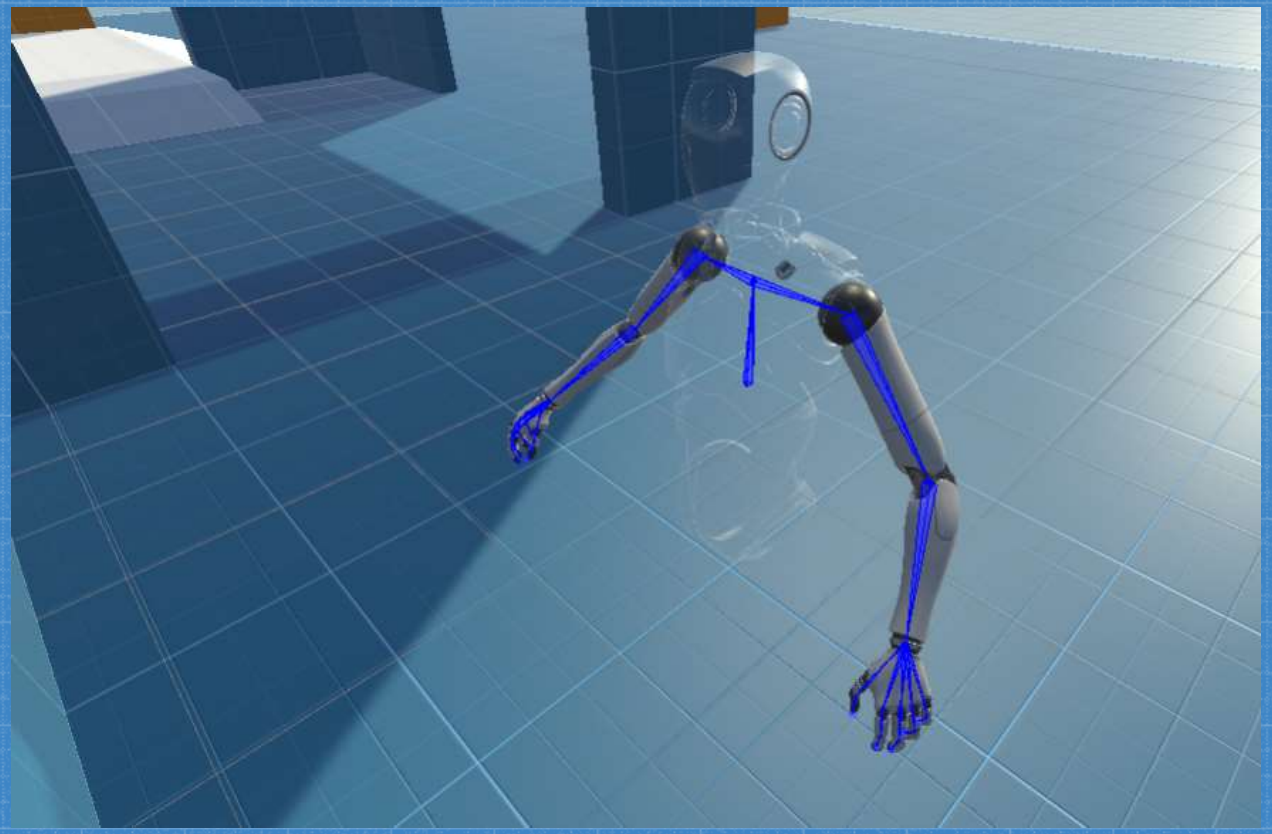
FEATURE – TASK ENVIRONMENT

- Loaded + setup through configuration
  - Consistency and repeatability
- Visual cues and task actions
  - Events broadcasted within the task
- Object location/orientation communication
  - Table for shadows to be cast on
  - Point light above
  - Materials on objects

- Virtual Arms Should Allow for Joint Modularity

Three Main Steps:

- Choose a Model
- Test Viability and Validate Model
- Develop Open Controls



FEATURE – VIRTUAL ARMS

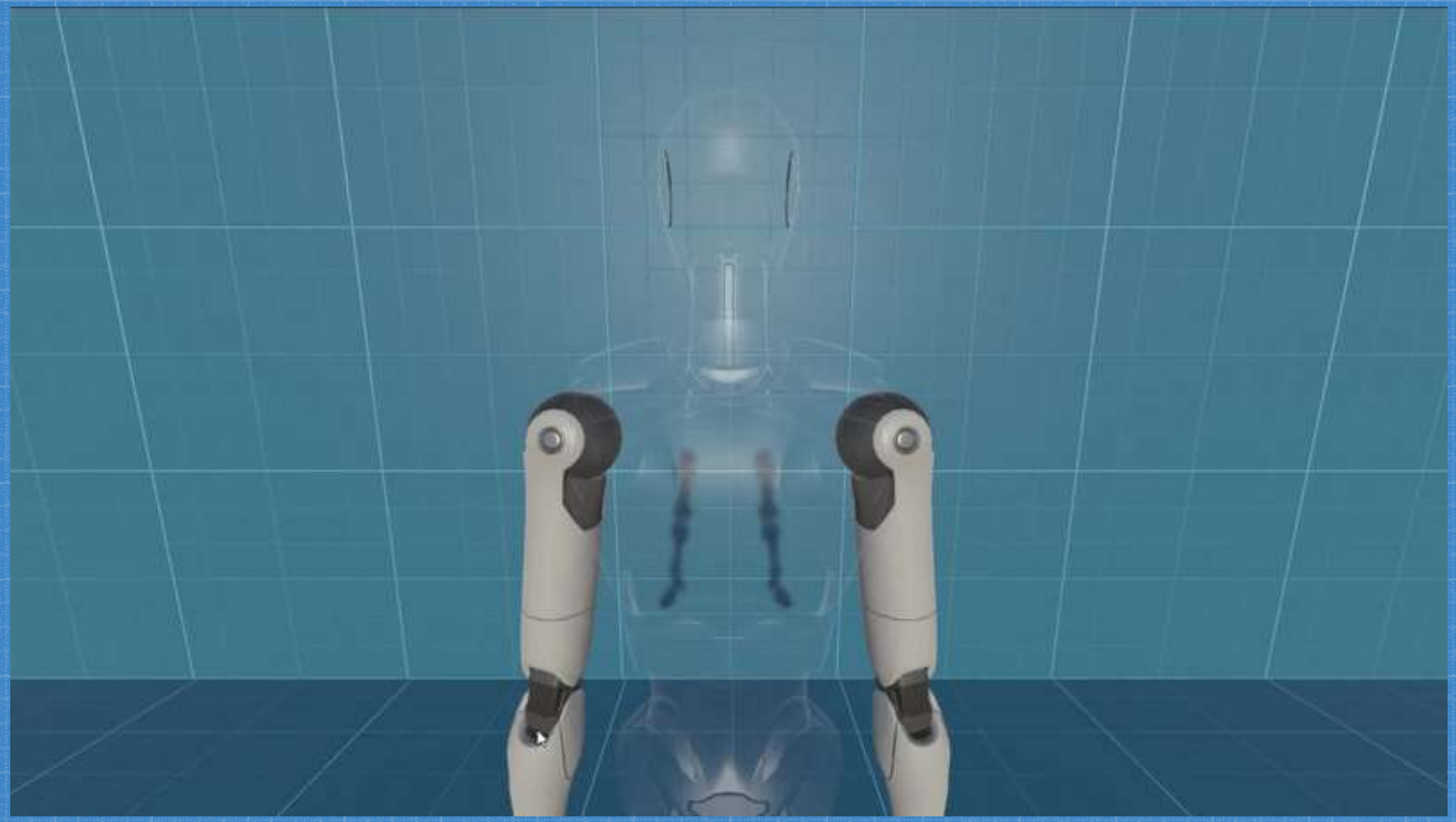
1) ADDUCTION

Recall:

- 2 Joints
- 5 Joints
- 7 Joints

What is Going On:

- Keyboard ‘A’ and ‘D’
- Rotation

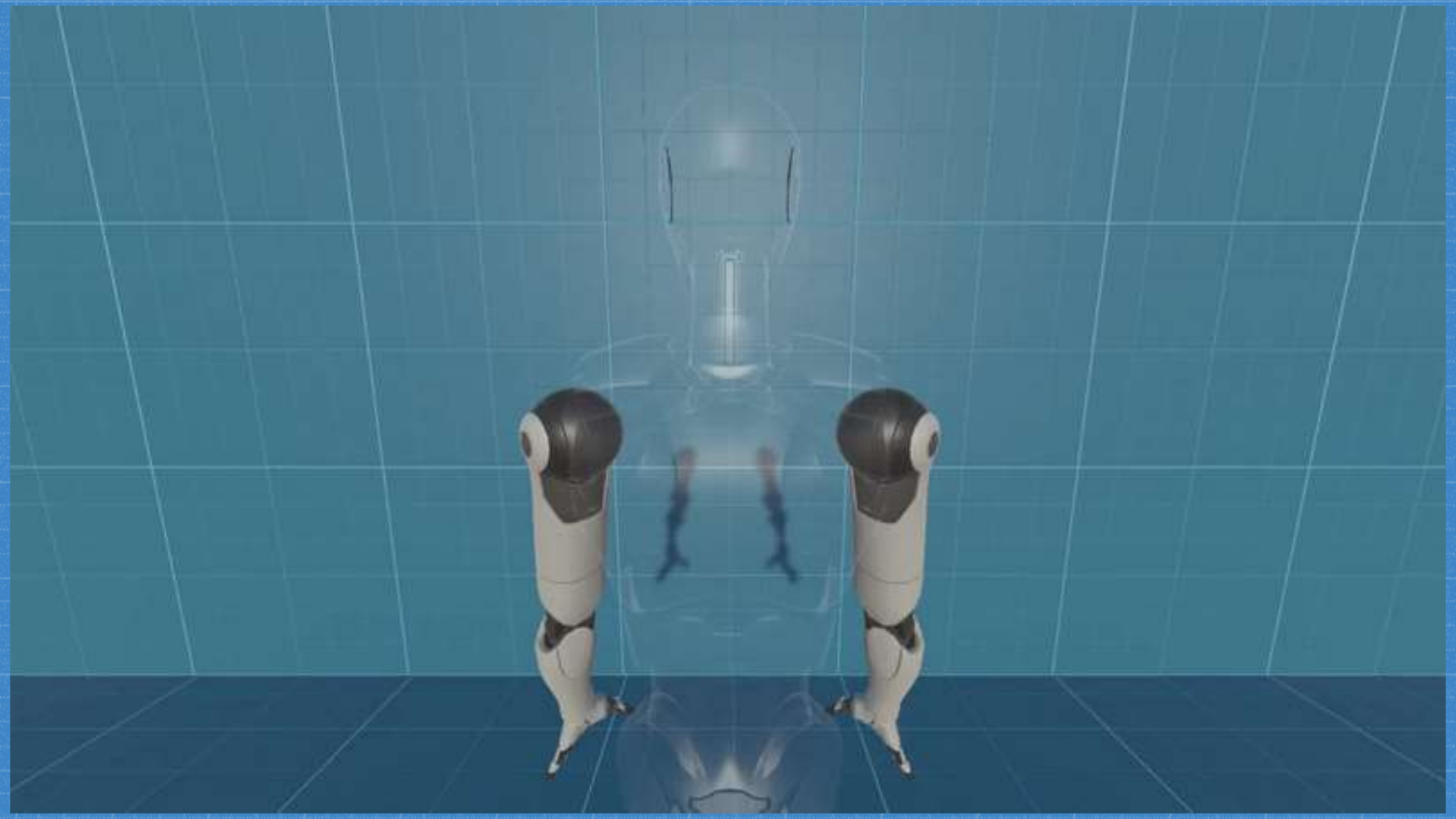


FEATURE – VIRTUAL ARMS

2) SHOULDER EXTENSION/FLEXION

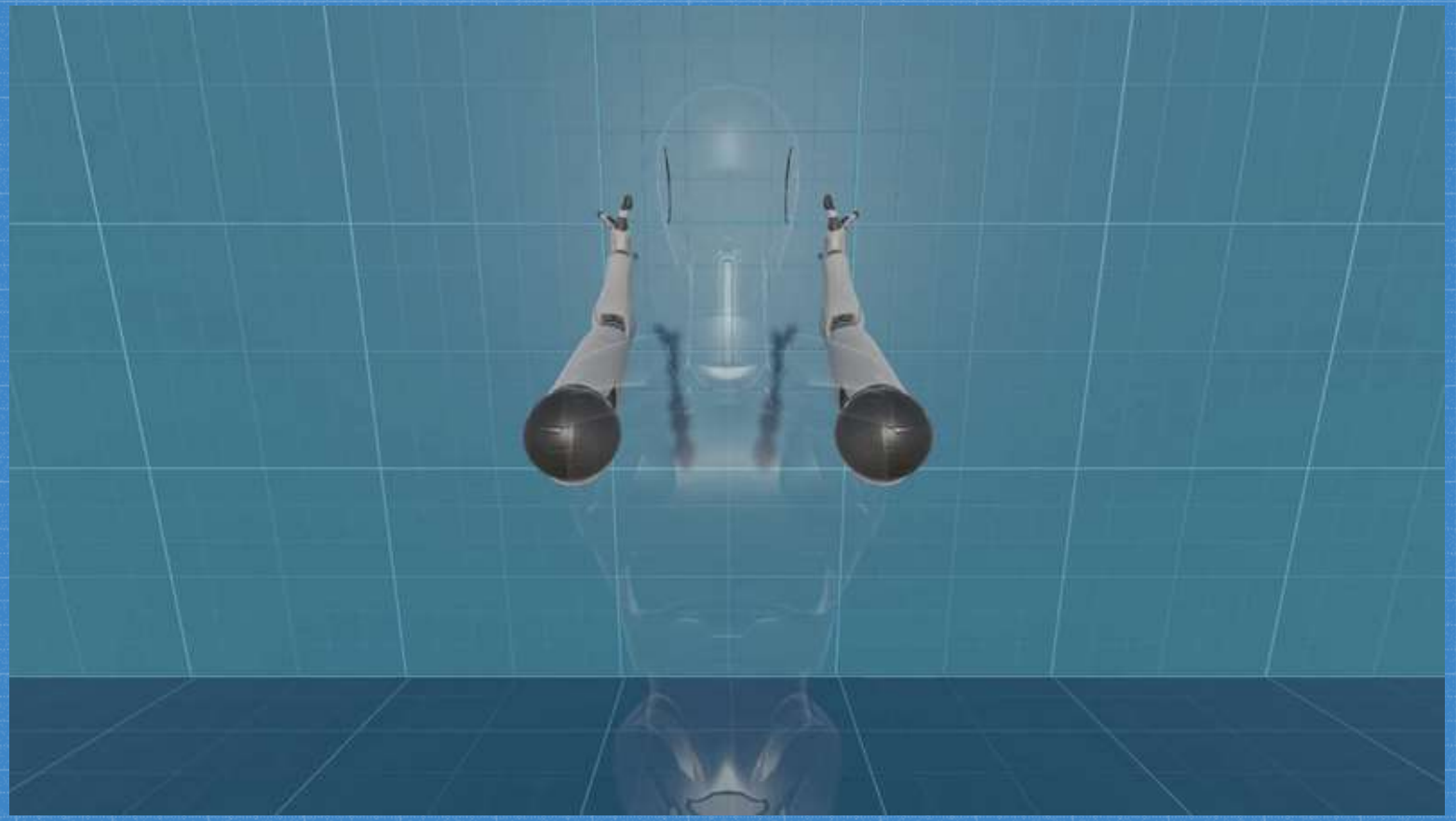
FEATURE – VIRTUAL ARMS

3) SHOULDER INTERNAL/EXTERNAL ROTATION



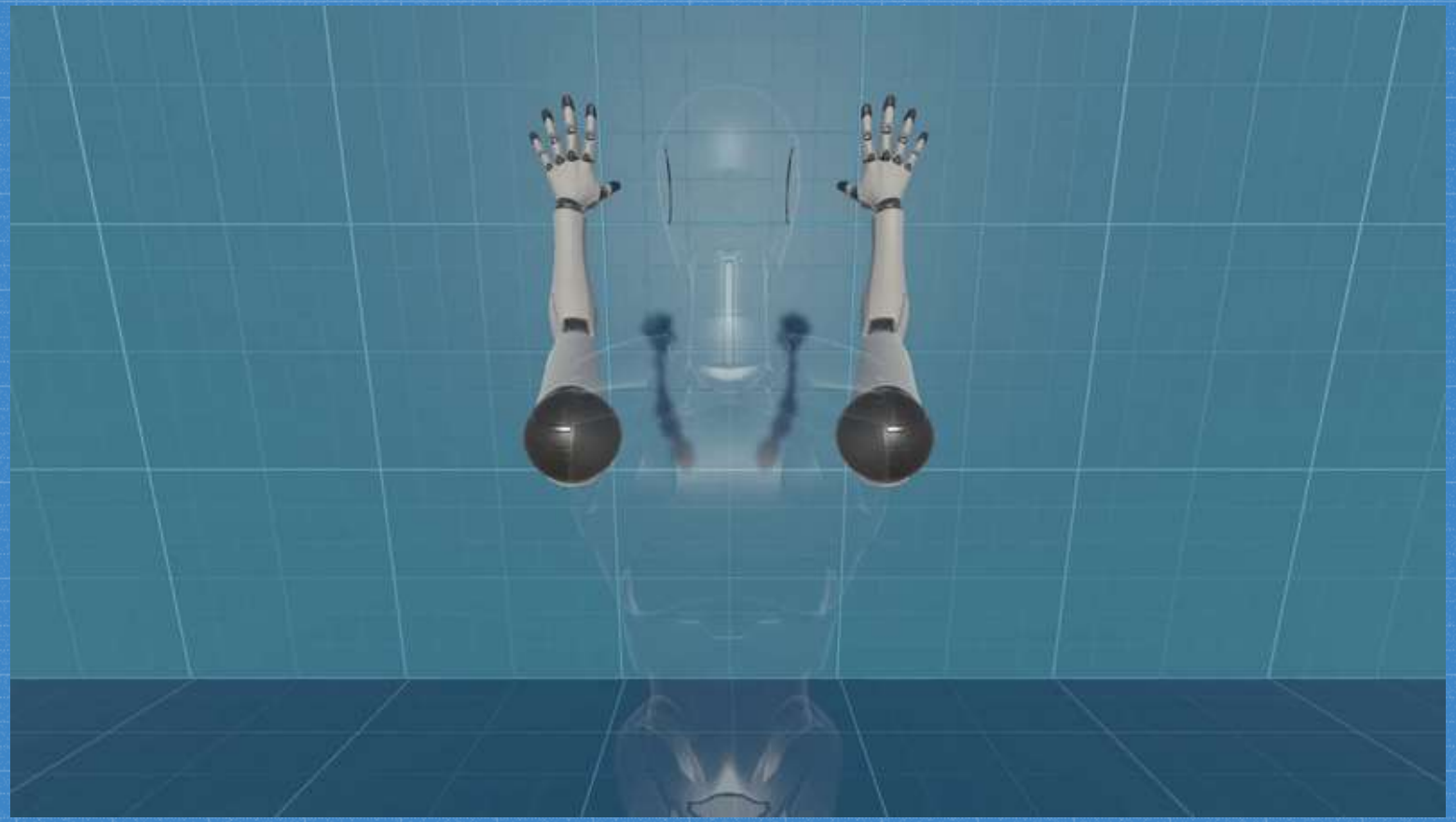
FEATURE – VIRTUAL ARMS

4) ELBOW FLEXION/EXTENSION



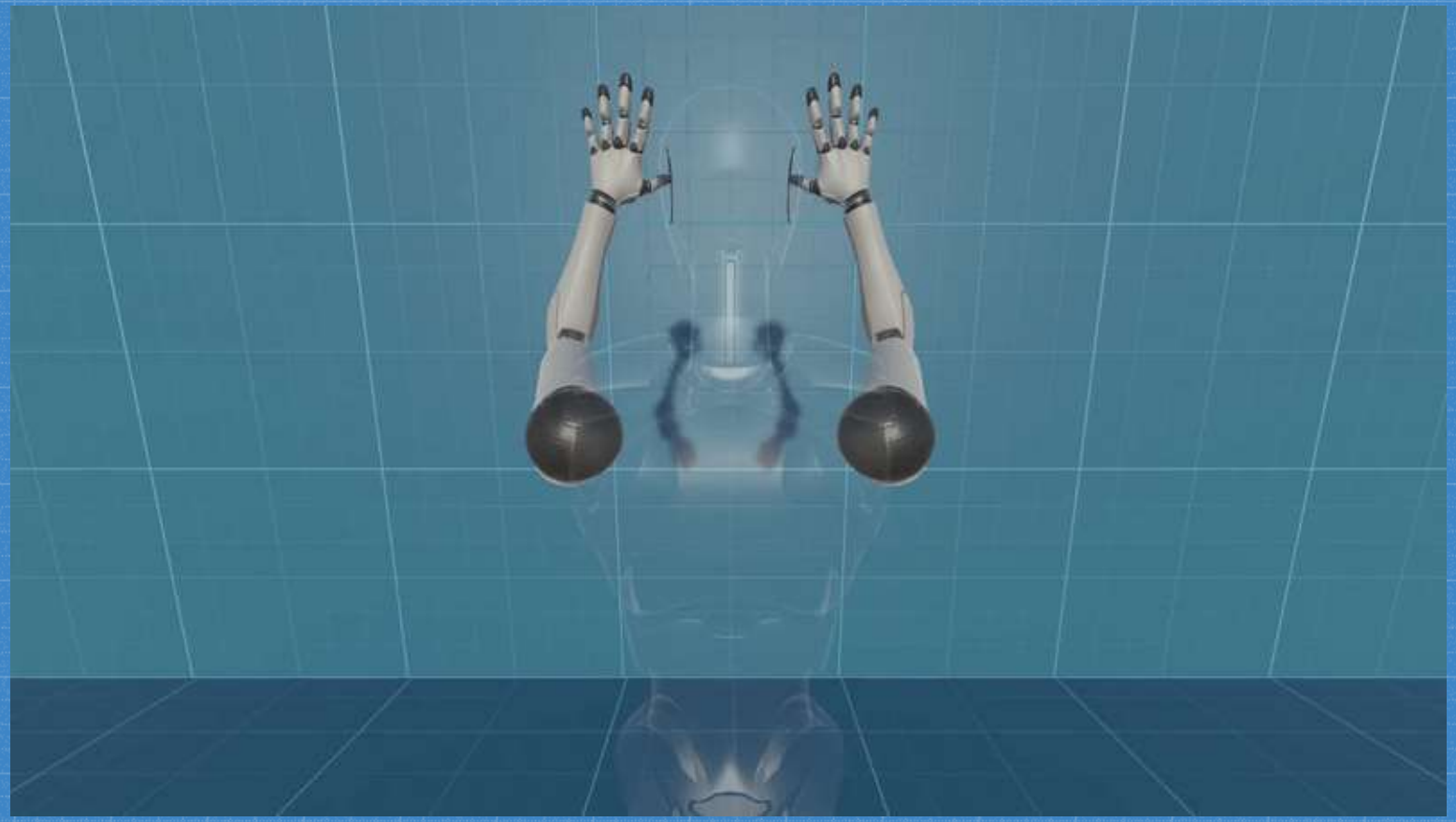
FEATURE – VIRTUAL ARMS

5) FOREARM PRONATION/SUPINATION



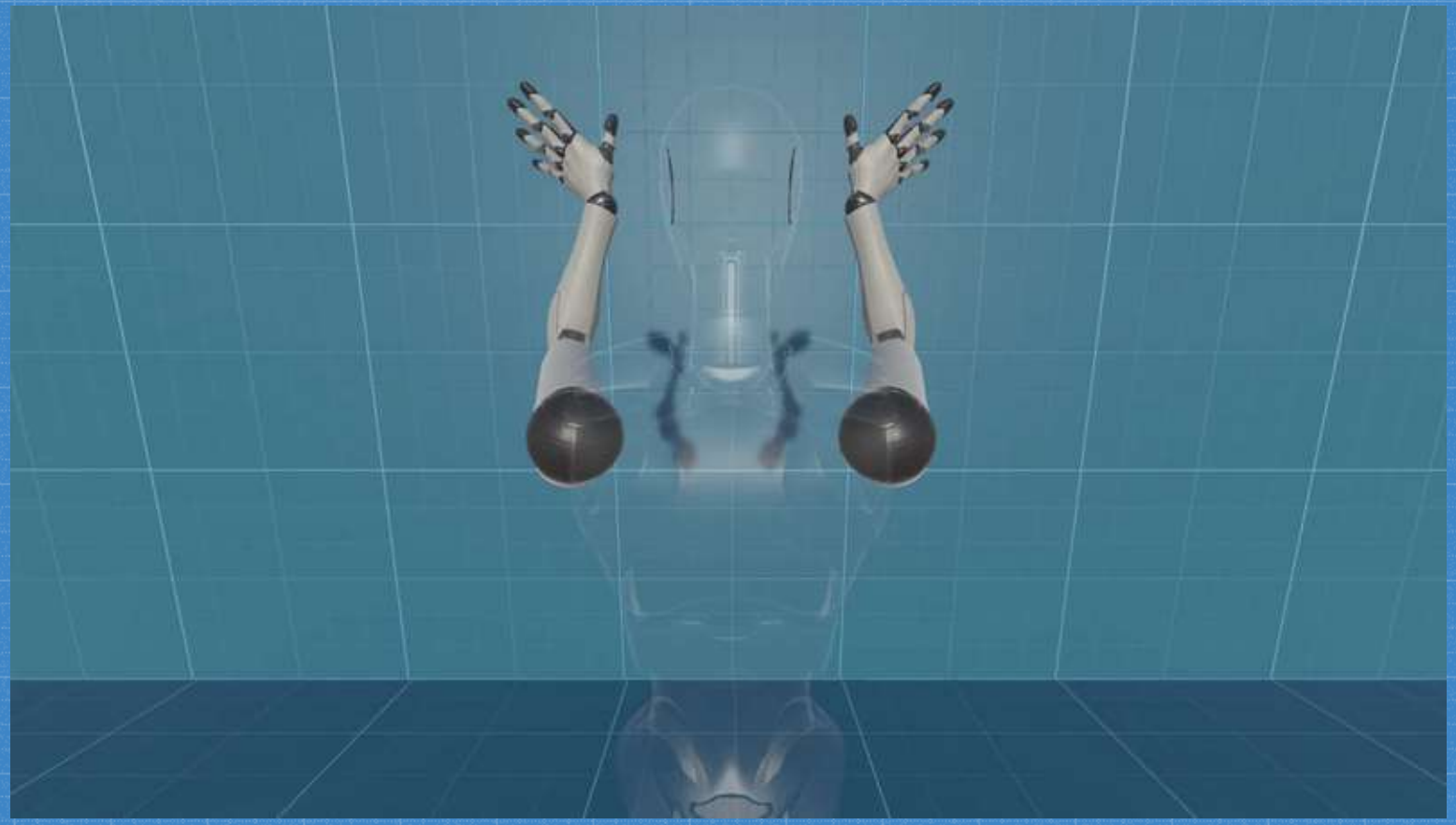
FEATURE – VIRTUAL ARMS

6) WRIST FLEXION/EXTENSION



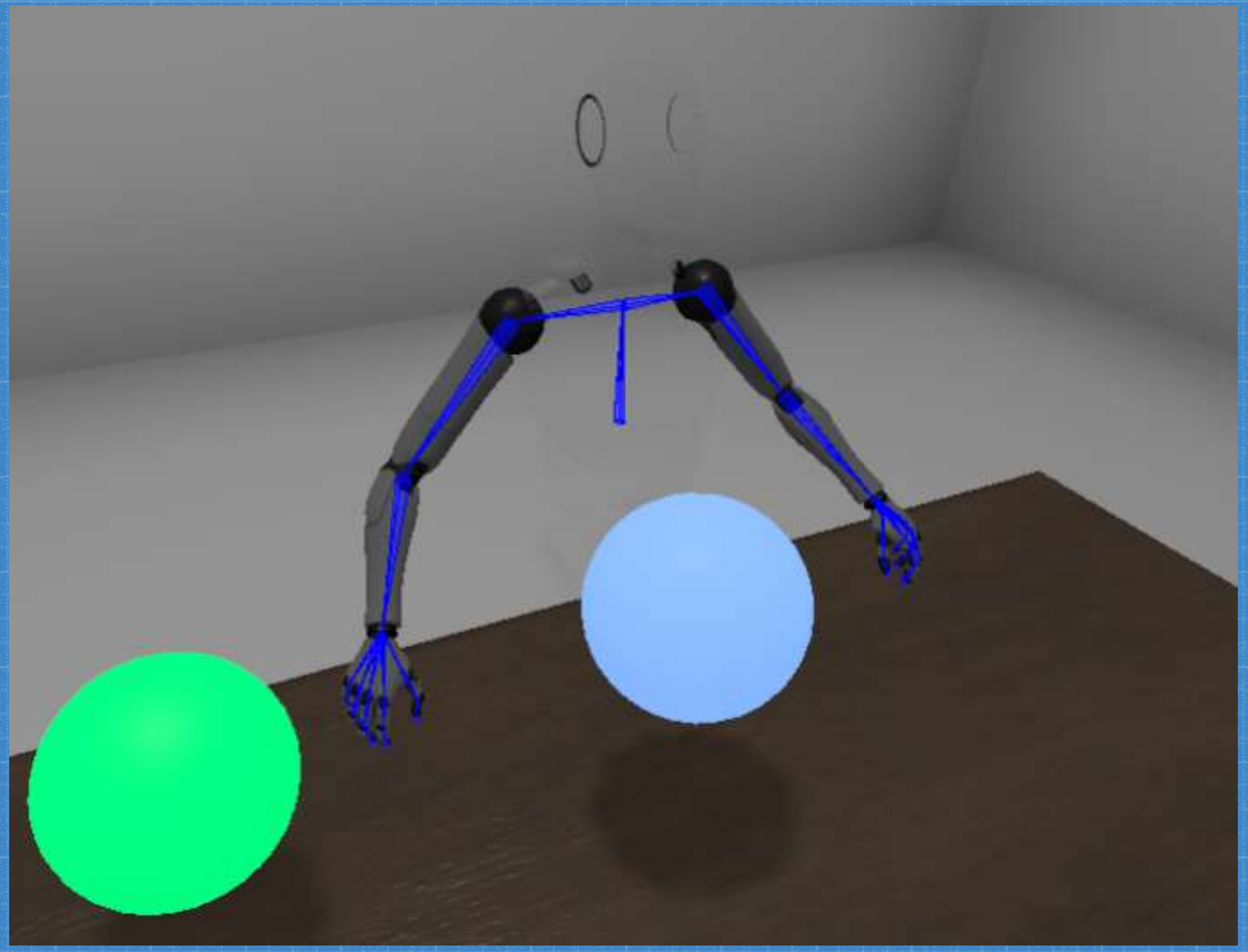
FEATURE – VIRTUAL ARMS

7) WRIST RADIAL/ULNAR DEVIATION



Last Points:

- Remaining Joints 8-15
- Capturing BLUE SABINO Outputs
- Tweaking the Arm Model



IDENTIFICATION OF POTENTIAL RISKS

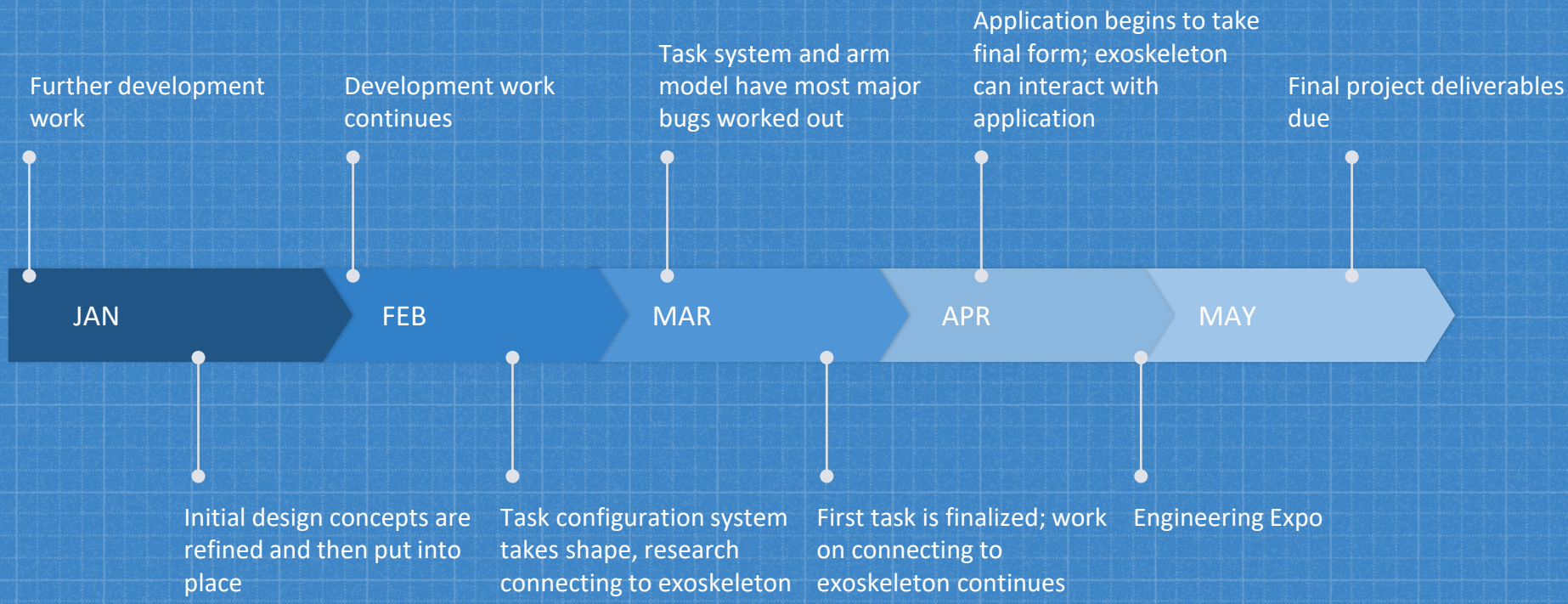
- The 3D environment should not move exoskeleton 
  - Only read data from exoskeleton

- Environment should not run below an acceptable framerate
  - Risk of breaking the accuracy of tests 

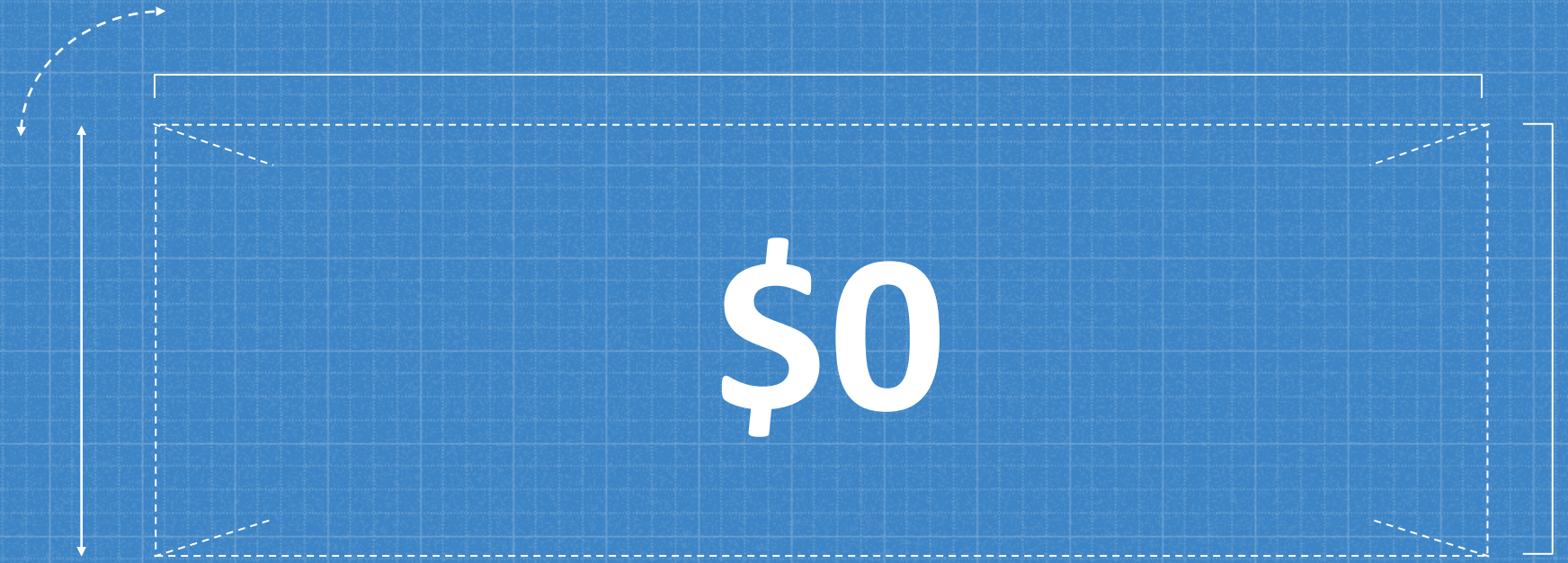
DESIGN VALIDATION PLAN

Requirement	Test	Test Subject	Target Date	Result	Recommendation
Virtual arms should allow modualar use of joints (listed below), but should move in a connected manner	Single point of control test (if the hand is moved (e.g. forward), the rest of the arm should move accordingly (i.e. in a realistic fashion)	Virtual arms	1/14/22	Rigged model's inverse kinematics acomplish the connected manner	Contact Jacob Wobido to see if we can acquire a model capable of having sections of the arm be transparent
Virtual arms should be controllable by the keyboard	Do the virtual arms respond correctly to the keyboard	Virtual arms	2/10/22	Virtual arms move by rotations about the x,y, and z axis for the upper arm, lower arm, and hand	Get more precise movement that match the joint requirements.
1) Adduction: shoulder out, arm away from body and towards body	Can the virtual arms perform the movement while staying connected	Virtual arms	2/24/22	Yes: works as intended	None
2) Extension and flexion: arm forward and backward	Can the virtual arms perform the movement while staying connected	Virtual arms	2/24/22	Yes: works as intended	None
3) Internal/external rotation about the shoulder	Can the virtual arms perform the movement while staying connected	Virtual arms	2/24/22	Yes: works as intended	None
4) Elbow flexion/extension	Can the virtual arms perform the movement while staying connected	Virtual arms	2/24/22	Behaves strangely but works	Tweak Vector3 numbers to be more precise. Very close
5) Forarm pronation/supination	Can the virtual arms perform the movement while staying connected	Virtual arms	2/24/22	This is accomplished about the hand so it does not take affect over the forearm as intended, however it works	None, as the model cannot twist the forearm without affecting the upper arm.
6) Wrist flexion/extension	Can the virtual arms perform the movement while staying connected	Virtual arms	2/24/22	Bare minimum movement is there, however it still does not look quite right.	Tweak Vector3
7) wrist radial deviation (thumb towards arm) and ulnar deviation	Can the virtual arms perform the movement while staying connected	Virtual arms	2/24/22	Movement is there, however it could be better	Tweak Vector3
Polish up the arms. Possibly add in joints 8) shoulder elevation and depression; 9) shoulder retration and protraction	Can the virtual arms perform the movements while staying connected	Virtual arms	3/11/22		
Additional joints 10-15 (The hand). 10 -11) Thumb; 12-13) Index; 14-15) Middle, Ring, Pinkie	Can the virtual hands perform the movement while staying connected	Virtual arms	3/25/22		
Simulink Simulaiton signal output connection.	Capture rotation matricies from simulaiton and use them to move the arms. Computation most likely needed	Lab setup with target computer and host computer running our Unity environment	TBD		
Menus should flow seamlessly together.	Set up a document that outlines a plan to traverse and validate each	Menus	End of Project	Menus should have no way to "get stuck". There should always be a way to get out of a scene without quitting the application.	Continue implementation of menu system.
Menus should be intuitive, with visual and potentially audio cues for UI elements.	Interact with menus, ensuring each UI element has distinct feedback features.	Menus, UI elements such as buttons, sliders, etc.	Duration of project development	Every UI element needs to have some sort of user feedback mechanism built in. Current progress reflects this requirement.	Use highlighting and pressing options to indicate if a UI element is being interacted with. Potential for auditory "clicks".
Menus need to be built in a modular way.	Menus use modular elements and introducing a new menu is seamless.	Menus, UI elements such as buttons, sliders, etc.	Duration of project development	Implementing new menus should be seamless and relatively straightforward. Current progress reflects this requirement	Use tools within Unity to help aid modularity.
Menus need to report the correct parameters to the data containers.	Pass data between menu front end and report what the data container is getting.	Menu, scriptable object.	Completion of task environment	Menus will assign the correct values to the parameters in the scene. Current progress DOES NOT reflect this requirement	Use data containers to pass parameters from front end menus into the scene seamlessly.
Each completed task will have its own configuration menu	For every new task implemented, a new configuration menu will be constructed to change various parameters.	Menus, UI elements such as buttons, sliders, etc.	Completion of project, further if necessary	As we are aiming to fully complete the Point to Point task, this may be a further development initiative after we finish our project	Use Unity tools that make building menus modular and seamless.
Tasks should be highly configurable	Change the given parameters for the point to point task. Ensure they are reflected	Point to point scriptable object	1/28/21	Correct values changed according to the changes given in parameters	Continue implementation of this with all tasks/within the system
Location of items should be clearly communicated on screen	Move the virtual arms to a given location on the screen. The movement should be intuitive	Goal object and virtual arm	2/2/21	Lighting and material of goal object did not allow for intuitive detection of location	Change the material on objects to have metallic or plastic properties so it may reflect lights and display position correctly
Oreintation of items should be clearly communicated on screen	Move the virtual arms to a given location and orientation on the screen. The movement should be intuitive	Goal object and virtual arm	2/2/21	Lighting and material of goal object did not allow for intuitive detection of orientation	Change the material on objects to have metallic or plastic properties so it may reflect lights and display position and orientation correctly.
Background of task area must be simple and not distract from the task	Perform any task, the background should not distract from the task	Point to point task	1/28/21	Background with grid lighting distracted from the indivudal objects	For now, implement the background as white with no special texture or lighting
Performance should not suffer	Run the unity profiler	3D Environment with profiler	2/18/21	Unity profiler reported above 200 frames/second in editor Will be even better on build.	None. All is good!
Point-To-Point Task Complete	Run through the point-to-point task start to finish	Point to Point task in whole 3D ENV	3/8/21		

PROJECT TIMELINE



BUDGET



$500 Allocated For The Project

THANK YOU

Any last questions, comments, concerns?