



Design Review 3

By: Landon Coonrod, Matthew Gardner, Peter Hilbert, & Karissa Smallwood

Client: Dr. Reza Razavian

Mentor: Karthik Srivathsan Sekar



Motivation & Impact

Clients Mission:

- Dr. Reza Razavian develops rehab-focused human-robot systems at NAU.
- Contribute to literature, and deepen our understanding of haptic interaction.

Why Haptics Matter:

- Robot can guide, assist, or resist movement
- Tracks how the body responds in real time
- Used in:
 - Stroke recovery (relearn reach & grasp)
 - Joint rehab (safe strength training)
 - Prosthetics training (controlled grip)
 - Balance training (fall-safe support)





Problem We're Solving

Old System:

- Not modular → every new study requires new code
- Hard to change tasks or parameters during a session
- Knowledge stuck in one developer's head
- Slow to test, slow to publish, not clinic-ready

Our Work:

- Modular control + experiment blocks
- Reuse behaviors instead of rewriting them
- Makes trials faster, safer, and scalable

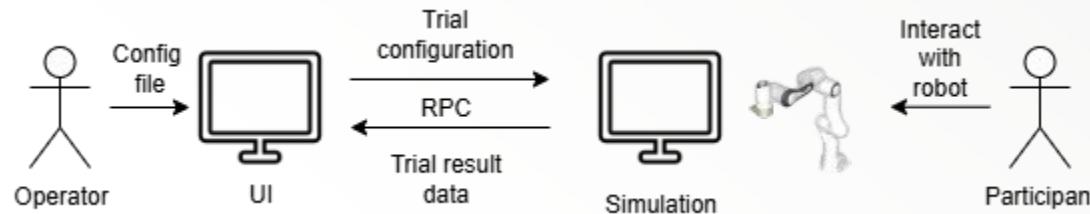




Solution Overview

Create a modular software system to support human-robot interaction experiments with the Franka Research 3 robot.

Problem	Solution Feature
No user-friendly interface.	Command-Line Interface for easy control.
No built-in data management.	Automated CSV/XML data storage.
No reusable experiment templates.	Modular architecture with XML configs.
Time-consuming setup.	Reusable, easily configurable experiments.





Key System Requirements

Domain & Functional Requirements:

- 1. Render 3D interactive simulations integrated with the Franka robot.**
 - Success: Smooth rendering with no visual glitches or frame drops during interaction.
- 2. Provide realistic, minimal delay haptic feedback.**
 - Success: Haptic control loop maintains ≤ 1 ms latency.
- 3. Support XML-based configuration files for experiment setup.**
 - Success: Setup time reduced by $\geq 50\%$.
- 4. Enable remote operation via a networked client-server connection.**
 - Success: Trials initialize within 10 seconds.
- 5. Automatically collect and organize experiment data.**
 - Success: All data saved within 3 seconds after trial completion.
- 6. Provide a user-friendly operator interface for experiment control.**
 - Success: Operators can complete 10+ consecutive trials without errors or crashes.



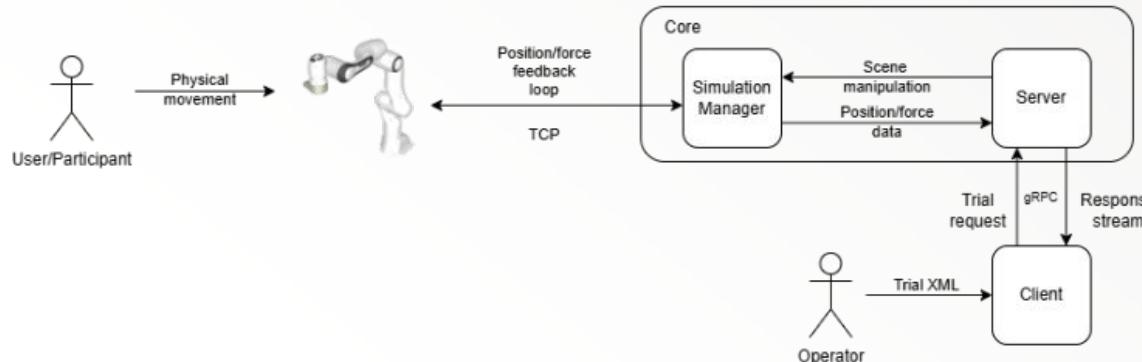


Implementation Details

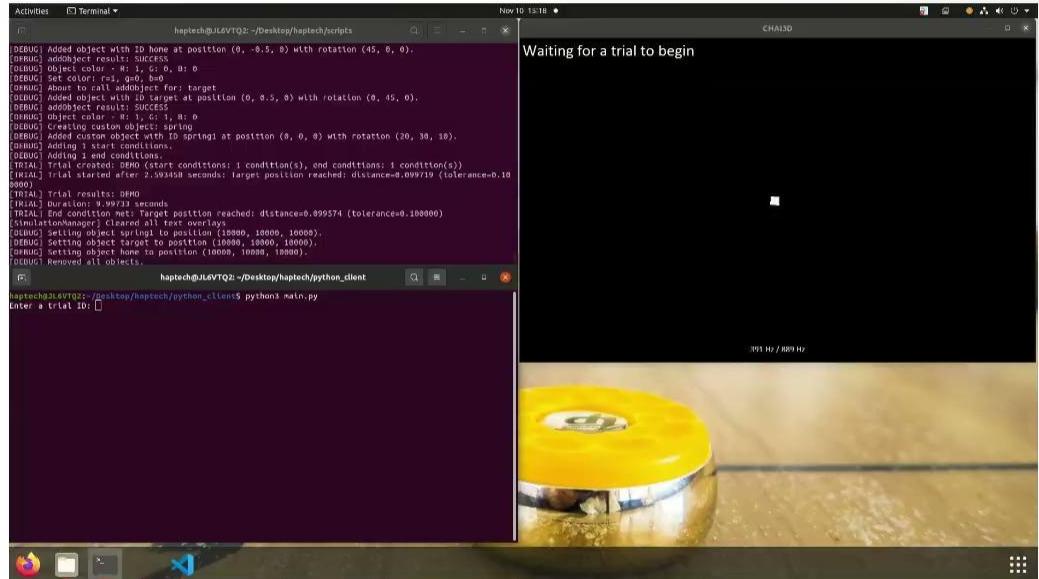
System Architecture

Main Components:

- **Client (UI)**: Loads XML configs, run trials, manages data.
- **Server**: Parses configs, manages trials, coordinates communication.
- **Simulation Manager**: Runs CHAI3D scenes and haptic rendering.
- **Franka Integration**: Bridges CHAI3D and robot via libfranka.



Prototype Review



Activities Terminal Nov 10 15:18 CHAI3D

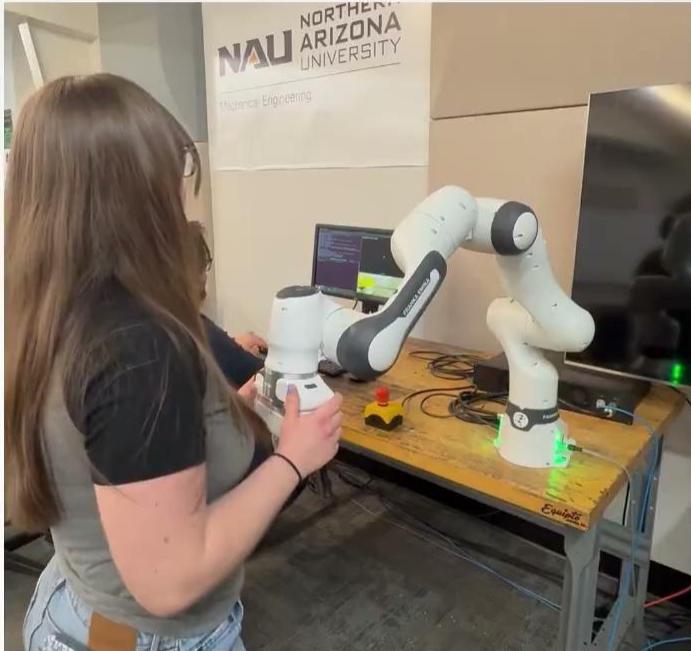
```
[DEBUG] Added object with ID home at position (0, -0.5, 0) with rotation (45, 0, 0).
[DEBUG] addObject result: SUCCESS
[DEBUG] Set color: r=1, g=0, b=0
[DEBUG] Set color: r=1, g=0, b=0
[DEBUG] About to call addObject for: target
[DEBUG] Adding target object with ID target at position (0, 0.5, 0) with rotation (0, 45, 0).
[DEBUG] addObject result: SUCCESS
[DEBUG] Object color = R: 1, G: 1, B: 0
[DEBUG] Creating custom object with ID Spring1 at position (0, 0, 0) with rotation (20, 30, 10).
[DEBUG] Adding i start conditions.
[DEBUG] Adding i end conditions.
[DEBUG] Adding i start conditions: 1 condition(s), end conditions: 1 condition(s)
[TRIAL] trial started after 2.393458 seconds: target position reached: distance=0.099719 (tolerance=0.10
000)
[TRIAL] Trial results: DPMO
[TRIAL] Duration was 0.99731 seconds
[TRIAL] End condition met: Target position reached: distance=0.099574 (tolerance=0.100000)
[DEBUG] Setting object target to position (10000, 10000, 10000).
[DEBUG] Setting object spring to position (10000, 10000, 10000).
[DEBUG] Setting object home to position (1000, 1000, 10000).
[DEBUG] Removed all objects.
```

hapttech@JL6VTQ2:~/Desktop/hapttech\$ python3 main.py

Enter a trial ID: □

Waiting for a trial to begin

391 Hz / 1091 Hz





Testing Plan

- **Unit Testing:**
 - **Frameworks:** PyTest for Python, Google Test (GTest) for C++
 - **Scope:** Core custom object and trial condition logic, simulation manager, XML parser
 - **Goal:** 100% unit tests passing, ~80% coverage of tested files
- **Integration Testing:**
 - **Scope:** Trial lifecycle, haptic feedback
 - **Goal:** Stable user workflow & realistic, responsive, and safe robot usage
- **Usability Testing:**
 - **Expert reviews:** Evaluate UI usability
 - **User studies:** Researchers and experiment participants simulate a real lab study environment using the tool
 - **Acceptance Testing:** Verify the product meets the expectations of Dr. Razavian and his team
 - **Goal:** Gather real user feedback in order to deliver a product that actually satisfies our client's initial needs and requirements



Testing Progress

- **Current Progress:**

- Unit testing frameworks installed and set up for the project.
- Integration testing performed throughout development with implementation of new features.
- High-level acceptance testing done with client throughout development.

- **Next steps:**

- Expand coverage of unit tests.
- More detailed integration testing of edge cases, stress testing.
- Usability testing with more participants.

```
peter@Peter:~/haptech/python_client$ python3 -m pytest
=====
platform linux -- Python 3.10.12, pytest-9.0.0, pluggy-1.6.0
rootdir: /home/peter/haptech/python_client
plugins: cov-7.0.0
collected 1 item

tests/test_parser.py .

=====
1 passed in 0.02s ===
peter@Peter:~/haptech/python_client$
```

```
peter@Peter:~/haptech/core/tests/build$ ./core_tests
[=====] Running 8 tests from 1 test suite.
[-----] Global test environment set-up.
[-----] 8 tests from TimeConditionTest
[ RUN ] TimeConditionTest.CreatesWithValidDuration
[ OK ] TimeConditionTest.CreatesWithValidDuration (0 ms)
[ RUN ] TimeConditionTest.EvaluatesFalseWhenTimeNotElapsed
[ OK ] TimeConditionTest.EvaluatesFalseWhenTimeNotElapsed (0 ms)
[ RUN ] TimeConditionTest.EvaluatesTrueWhenTimeElapsed
[ OK ] TimeConditionTest.EvaluatesTrueWhenTimeElapsed (0 ms)
[ RUN ] TimeConditionTest.EvaluatesTrueWhenTimeExceeded
[ OK ] TimeConditionTest.EvaluatesTrueWhenTimeExceeded (0 ms)
[ RUN ] TimeConditionTest.HandlesZeroDuration
[ OK ] TimeConditionTest.HandlesZeroDuration (0 ms)
[ RUN ] TimeConditionTest.HandlesInvalidDuration
[TimeCondition] Invalid duration_seconds parameter: invalid_number
[ OK ] TimeConditionTest.HandlesInvalidDuration (0 ms)
[ RUN ] TimeConditionTest.GeneratesCorrectMessage
[ OK ] TimeConditionTest.GeneratesCorrectMessage (0 ms)
[ RUN ] TimeConditionTest.DemonstrateMockUsage
[ OK ] TimeConditionTest.DemonstrateMockUsage (0 ms)
[-----] 8 tests from TimeConditionTest (0 ms total)

[-----] Global test environment tear-down
[=====] 8 tests from 1 test suite ran. (0 ms total)
[ PASSED ] 8 tests.
peter@Peter:~/haptech/core/tests/build$ |
```

Segmentation Faults

Issues: CHAI3D, Simulation Manager, client/server (gRPC).

Solution: Used debugging tools.

Impact: No more segmentation faults.

Hardware Performance

Issues: Real-time & rendering saturated hardware.

Solution: Limit loops, less logging, threads.

Impact: Stable running across computer.

Runtime Crashes

Issues: Mixed source crashing.

Solution: Added debugging, proper shutdowns, code review.

Impact: No more crashes.



Project Schedule

- Phase 1 - Main Integration
 - Core components of codebase
- Phase 2 - Polishing Steps (Current)
 - Optimization
 - Error Handling
- Phase 3 - Documentation
 - User Manual
 - Documentation in code

PHASE WEEKS	PHASE 1						PHASE 2				PHASE 3					
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
TASK TITLE																
Development																
Set Up Prototype with Franka Robot	●															
Add Support for Built in Objects		●														
Investigate Performance Issues			●													
Research Custom Objects in CHAI3D & Create a Basic Trial XML Config File				●												
Fix Segmentation Fault Errors					●											
Design Interface for Custom Objects						●										
Implement Custom Object Interface							●									
Add data from scene objects								●								
Infrastructure for trial end conditions									●							
Trial Start on Home Position										●						
Home robot for each trial											●					
Create End Conditions											●					
Refinement and Debug												●				
Optimize Performance													●			
Create Debug Messages													●			
Error Handeling														●		
Project Documentation														●		
Use Manual														●		
Code Documentation														●		



Conclusion

Recap:

- Haptic technology is key part of advancing human-robot interaction
- Our clients current setup is limited by lack of modularity, poor accessibility, and an inefficient workflow.
- Our proposed solution is a command line user interface with a reusable experiment set up and data management.

Next Steps:

- Final optimizations, and small additions.
- Document the project and include an user manual.
- We are confident and excited to deliver a meaningful and impactful solution that helps advance the technology of Human Robot Interaction



• • • • •

Thank You!

Any questions?

• • • • •