

SPRS-24 Feature Document

User should be able to control the python client using a Controller to control the car and game hud

Name: [Nicolas Gonzalez](#)

Team Member(s): [Nicolas Gonzalez](#) , [Emmanuel Perez](#)

Project: SPARSE

Product Owner(s): [Miguel Alonso Jr](#)

Mentor(s):

Instructor: [Masoud Sadjadi](#)



SPRS-24 - User should be able to control the python client using a Controller to control the car and game hud DONE

Description

As a User I should be able to control the python client using a Controller to control the car and game hud.

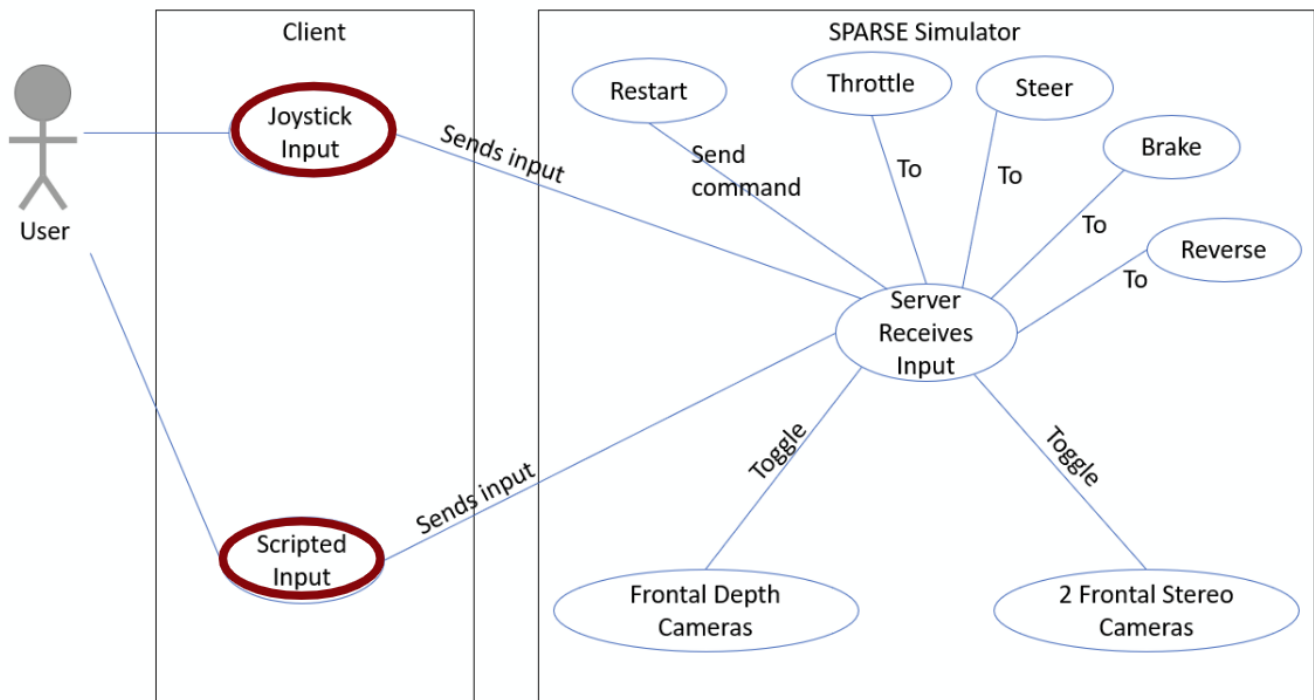
Acceptance Criteria

- Be able to make gamepad input into python client
- Switch between controller mode and code mode
- All functions are mapped to a gamepad input

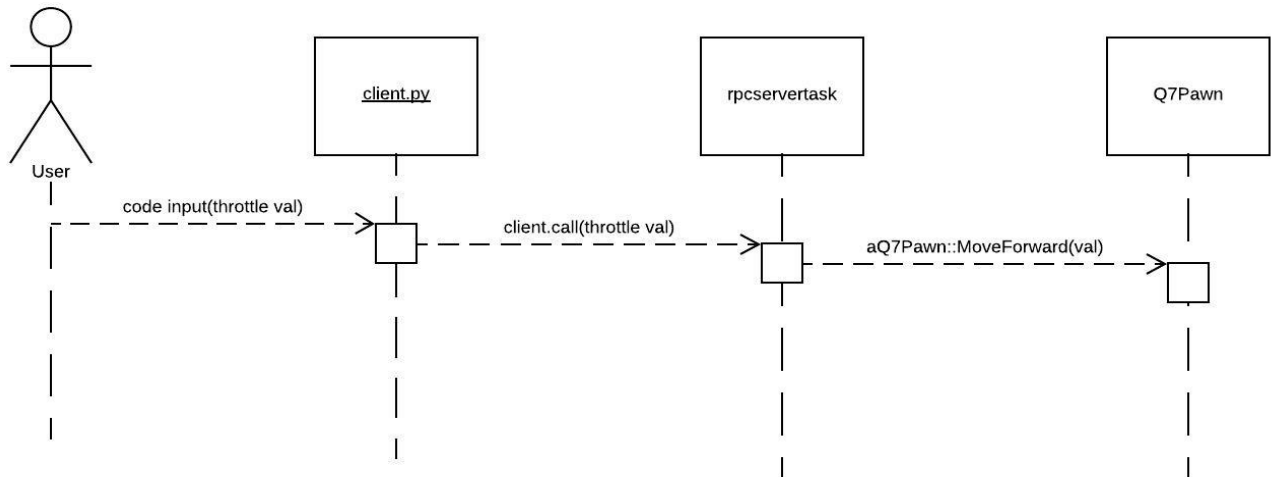
Use Case

- Name: Control sim with gamepad
- Actor: AI developer
- Preconditions: Unreal compiles and runs
- Description <Flow of events>: User can control all SPARSE function with both code and a gamepad

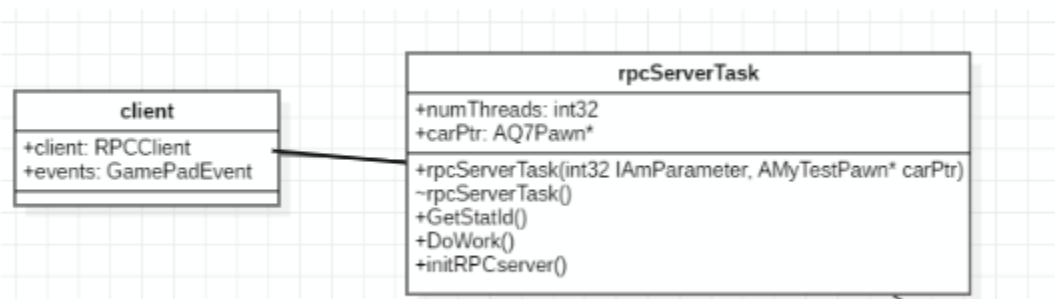
Use Case Diagram



Sequence Diagram



Class Diagram



Unit Test

- Test case ID: SPRS-24-A
- Description/Summary of Test: Test Gamepad Steering using left stick
- Pre-condition: Unreal sim runs and rpc connection
- Expected Results: User can steer with controller
- Actual Result: User can steer with controller
- Status (Fail/Pass): PASS
- Test case ID: SPRS-24-B
- Description/Summary of Test: Test Gamepad throttle using right trigger
- Pre-condition: Unreal sim runs and rpc connection
- Expected Results: User can throttle with controller
- Actual Result: User can throttle with controller
- Status (Fail/Pass): PASS
- Test case ID: SPRS-24-C
- Description/Summary of Test: Test Gamepad Reverse using left trigger
- Pre-condition: Unreal sim runs and rpc connection
- Expected Results: User can Reverse with controller

- Actual Result: User can Reverse with controller
- Status (Fail/Pass): PASS
- Test case ID: SPRS-24-D
- Description/Summary of Test: Test Gamepad Handbrake using PlayStation X button
- Pre-condition: Unreal sim runs and rpc connection
- Expected Results: User can brake with controller
- Actual Result: User can brake with controller
- Status (Fail/Pass): PASS
- Test case ID: SPRS-24-E
- Description/Summary of Test: Test Gamepad reset using start button
- Pre-condition: Unreal sim runs and rpc connection
- Expected Results: User can reset with controller
- Actual Result: User can reset with controller
- Status (Fail/Pass): PASS
- Test case ID: SPRS-24-F
- Description/Summary of Test: Test Gamepad camera toggle on/off with PlayStation Triangle button
- Pre-condition: Unreal sim runs and rpc connection
- Expected Results: User can camera HUD toggle on/offwith controller
- Actual Result: User can camera HUD toggle on/off with controller
- Status (Fail/Pass): PASS
- Test case ID: SPRS-24-G
- Description/Summary of Test: Test Gamepad codemode toggle using options button
- Pre-condition: Unreal sim runs and rpc connection
- Expected Results: User can switch to code mode with controller
- Actual Result: User can switch to code mode with controller
- Status (Fail/Pass): PASS

Visual User Guide

<like one or two screenshots of the feature. For the hardware project, a photo of device is required>

```
C:\Users\nicog\Desktop\Testmaster\sparse-v1\python>python client.py
CONNECTED
Entering Controller Mode
Key Press: BTN_SELECT 1
CONNECTED
Entering Code Mode
-----
To Run Test Script: 'testscript'
To Throttle: 'throttle [0:1]'
To Reverse: 'reverse [0:1]'
To Steer: 'steer [-1:1]'
To Engage Handbrake: 'brakeon'
To Disengage Handbrake: 'brakeoff'
To Toggle Cameras: 'toggleFcameras'
To Reset Simulation: 'reset'
To Re-Enter Controller Mode: 'controller'
-----
```

