*Florida International University*

*School of Computing and Information Sciences*

Software Engineering Focus

Feature Document

User Story ID #752 Video Game Controller

**Name: Pachev Joseph**

**Team Member(s): Bernardo Pla, Daniel Rivero, Daniel Khawand,**

**Project: WebVR 1.0**

**Product Owner(s)**:

**Mentor(s)**: Francisco Ortega

**Instructor**: Masoud Sadjadi

**User Story Name: #752 Design Connection For Mouse**

* As a developer, I would like a modular interface to connect a video game controller to any computer and have it be usable.

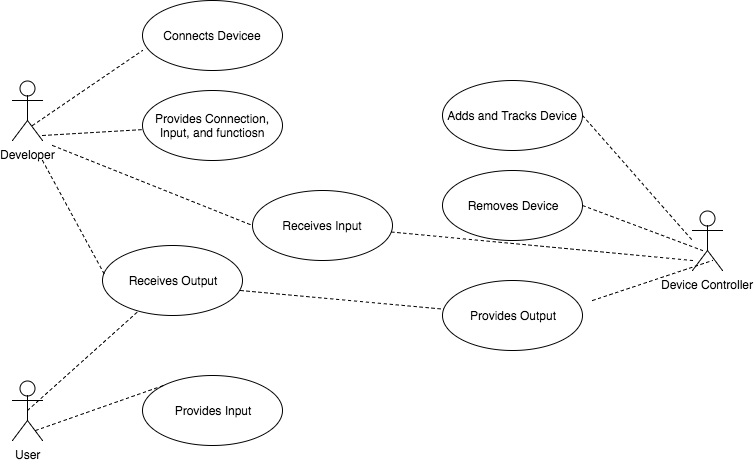
Acceptance Criteria

* Game controller can be read from device
* Game controller is able to accept input and turn it into output for developers to use
* Game controller format is outputted in JSON to allow developers to easily read and modify

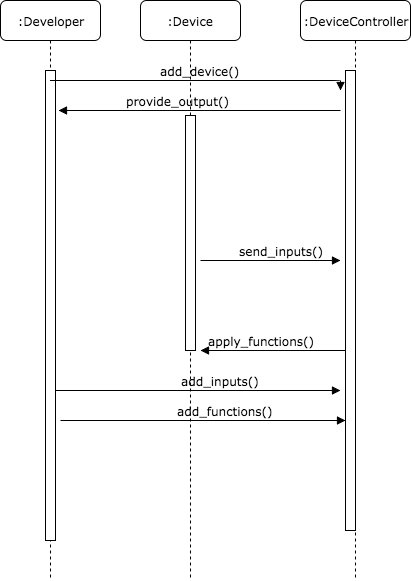
**Use Case**

* Name: Video Game Controller
* Actor: Developer, User, Device Controller
* Preconditions: Webvr-input has started and is listening for devices
* Description :
  + Developer connects controller
  + Developer provides which inputs and functions are needed
  + Device controller reads and adds controller
  + User provides output
  + Device controller receives input and provides appropriate output to developer.

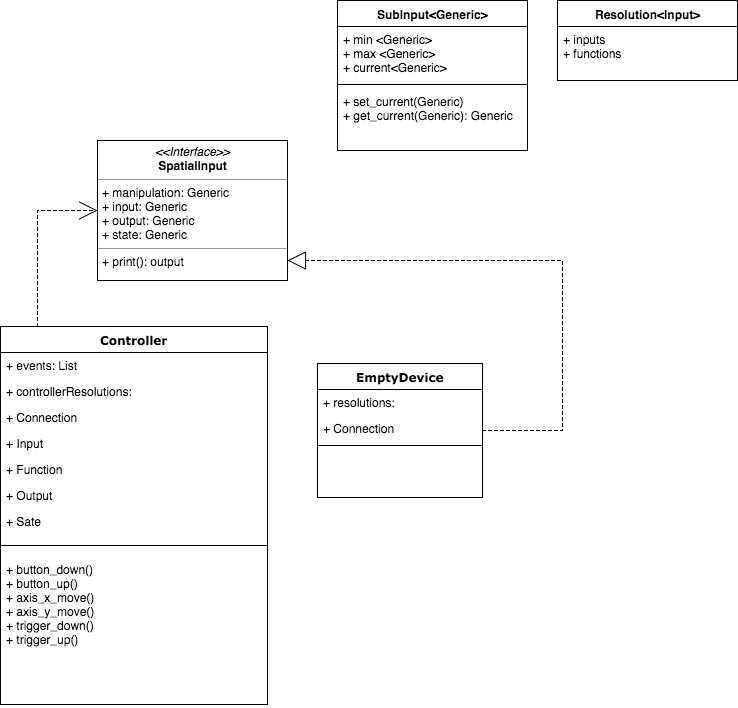
**Use Case Diagram**



**Sequence Diagram**



**Class Diagram**



**Test case ID: WV-CO-IN012: Sunny**

* **Description/Summary of Test:** The developer will launch the demo class and only provide  
  Inputs for left analog and the ‘A’ button
* **Pre-condition:** The main class in the rust library is running. A controller must be connected to the machine and is fully operational
* **Expected Results:** Once binary is running, only the left analog and ‘A’ button will provide output to developer.
* **Actual Result:** The developer only sees the result of ‘Button A down’ and ‘LeftY moved’
* **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>

