

# SmartPeerJS: A library for turning smartphone in interactive controller

---

EMMA POLIAKOVA

# Motivation

- Almost everyone owns a smartphone with good processing power and an array of sensors
- Touchscreen for various game controllers – joystick, NES/Playstation style controller, camera for hand tracking or object recognition, accelerometer for racing games
- Smartphone controllers are free and require no software installation



Leap motion controller



Makey makey invention kit

# Goals

---

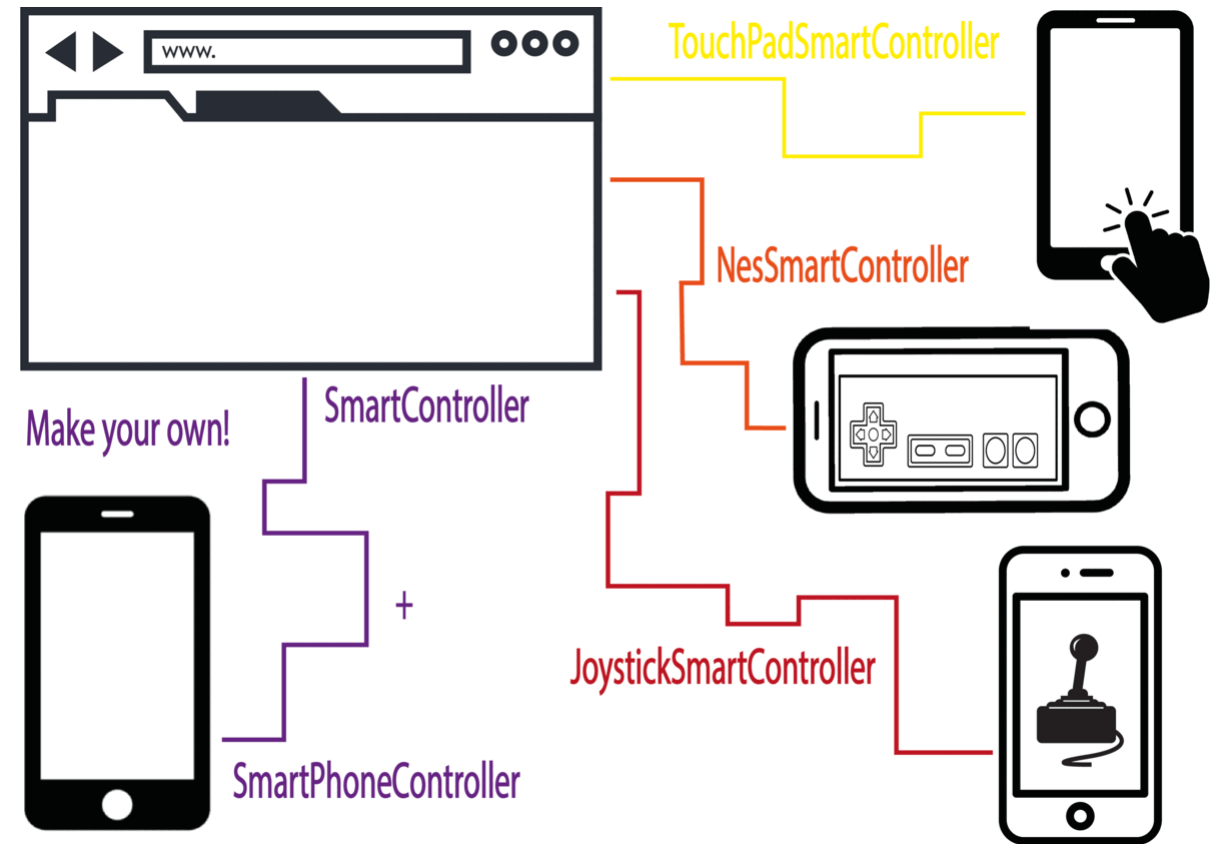
- Automatic connection handling
- Supporting multiplayer
- Hosting pre-made controllers
- Easy way of creating new controllers
- Creating a GitHub organization for the library
- Making user documentation, tutorials and demos
- Publishing the library on npm



SmartController organization logo

# Design

- Survey to name the library, suggestions included web smartphone controller, game controller, rtc smart controller, JS P2P smartphone game
- Use WebRTC and PeerJS to connect smartphone via QR code
- SmartController manages the connection in the computer browser , SmartPhoneController is responsible for the smartphone side



Simplified SmartController structure

# Implementation

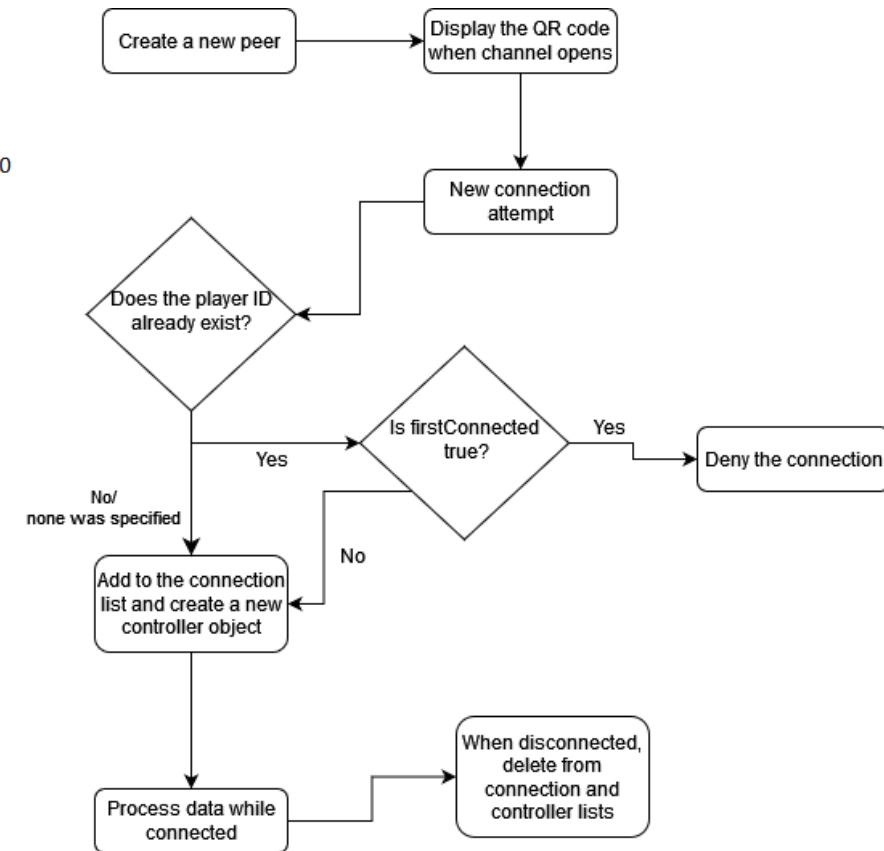
- Use of PeerJS



- Qr code method

<https://smartcontrollerjs.github.io/Controllers/joystick.html?id=1234567&playerid=1&throttle=20>

- Events with EventEmitter2
- Statistics – ping, user message rate, statistics message rate
- SmartPhoneController send method and message throttle





Basic SmartController flowchart

# Working with users

- While developing the library I worked with two fellow students who used it to create their own projects
- There were weekly meetings where we discussed and issues or problems they had, or any functionalities that they needed but the library did not offer yet
- At the end I conducted a short interview to find out their overall experience of working with the library

Smart Controller

[Introduction](#)  
[Getting Started](#)  
[Smart Controller](#)  
[Statistics](#)  
[Controllers](#)   
[Smart Phone Controller](#)  
[Demos](#) 

This site uses [Just the Docs](#), a documentation theme for Jekyll.

Search Smart Controller

[Smart Controller on GitHub](#)

## Getting started

### Dependencies

SmartController uses [peerjs](#) to provide peer to peer connection between PC and smartphone browsers. The events are handled by [EventEmitter2](#) and the qr codes are generated with [EasyQRCodeJS](#)

### Include the library



install NPM package: `npm install smartcontroller`

```
// The usage -  
import "smartcontroller";
```

html file:

```
```html  
<script src="https://unpkg.com/smartcontroller@3.2.1/dist/smartcontroller.min.js"> </script>  
```
```

Smart Controller

[Introduction](#)  
[Getting Started](#)  
[Smart Controller](#)  
[Statistics](#)  
[Controllers](#)   
[Smart Phone Controller](#)  
[Demos](#)   
[Multiplayer touchpad tutorial](#)  
[Nes controller tutorial](#)

This site uses [Just the Docs](#), a documentation theme for Jekyll.

Search Smart Controller


[Smart Controller on GitHub](#)

[Demos](#) / [Nes controller tutorial](#)

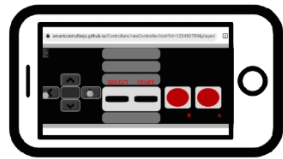

## NES controller tutorial

Try the demo!

Scan the QR code to connect!  
Peer Id not ready.

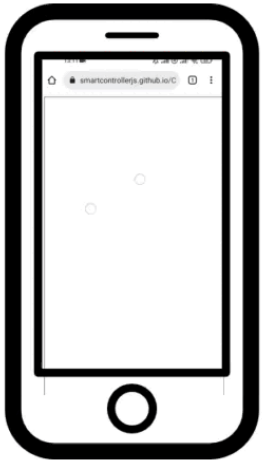


This tutorial will show you how to create a simple website [demo](#) to process data from NES controller. Everytime an arrow key is pressed it is highlighted on the screen. [Source code](#) and [html file](#).

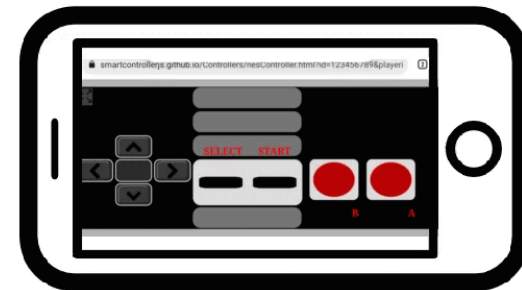
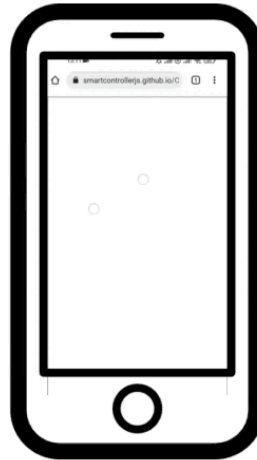
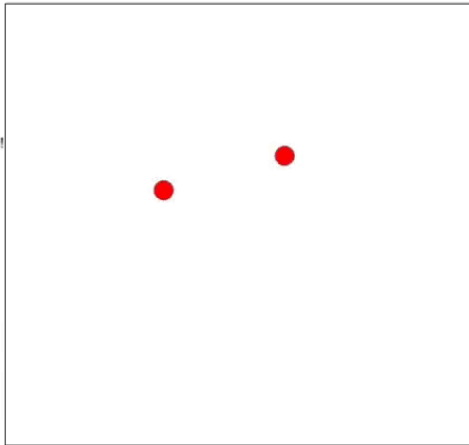


# Demos

- Showcase the library functionalities
- Tutorials for new users



Scan the QR code to connect!



Thank you for your attention!

---