

# GESTOS COMPLEJOS

## CONCEPTOS BÁSICOS

Vamos a añadir la interpretación de diferentes gestos. Para eso vamos a usar la librería hammer, que nos permite enlazar de una manera muy sencilla diferentes gestos con acciones.

En concreto vamos a detectar la doble pulsación, la presión, es swipe y el girar.

Habilitaremos un div para mostrar los efectos. Todas las animaciones las haremos con CSS

## PRIMEROS PASOS

Descargamos la librería hammer y la colocamos dentro de js:

<http://hammerjs.github.io/>

Añadimos un div para reflejar los gestos e incluimos la librería:

```
<body>
  <button id="claro">CLARO</button>
  <button id="oscuro">OSCURO</button>
  <div id="zona-gestos"></div>

  <script type="text/javascript" src="cordova.js"></script>
  <script type="text/javascript" src="js/fastclick.js"></script>
  <script type="text/javascript" src="js/hammer.min.js"></script>
  <script type="text/javascript" src="js/gestos.js"></script>
</body>
```

Añadimos estilos para el div y dos animaciones:

```
#zona-gestos{
  background-color: #7fba67;
```

## PHONEGAP (I)

```
width:100%;

min-height:300px;

margin-top: 10px;

}

#zona-gestos.doubletap{

  -webkit-animation: pulsacion 1.5s 1;

}

#zona-gestos.press{

  -webkit-animation: presion 1.5s 1;

}

@-webkit-keyframes pulsacion{

  0%,100% {

    background-color: #7fba67;

  }

  50%{

    background-color: #8de1bd;

  }

}

@-webkit-keyframes presion{

  0%,100% {

    background-color: #7fba67;

  }

  50%{

    background-color: #577a76;
```

```
}  
}
```

Añadimos el código para gestionar los gestos:

```
iniciaHammer: function () {  
    var zona = document.querySelector('#zona-gestos');  
    var miHammer = new Hammer(zona);  
  
    zona.addEventListener('webkitAnimationEnd', function () {  
        zona.className = "";  
    });  
    miHammer.on('doubletap', function () {  
        zona.className = "doubletap";  
    });  
    miHammer.on('press', function () {  
        zona.className = "press";  
    });  
},
```

Y lo llamamos al cargar:

```
if ('addEventListener' in document) {  
    document.addEventListener('DOMContentLoaded', function () {  
        FastClick.attach(document.body);  
        app.inicio();  
    });  
}
```

## PHONEGAP (I)

```
    app.iniciaHammer();  
  }, false);  
}
```

Ahora debería responder a los eventos de doble click y pulsar.

## SWIPE

Vamos a añadir la detección de swipe.

Primero creamos las animaciones que simularán que arrastramos la caja central:

```
#zona-gestos.swipe-derecha{  
  -webkit-animation: swipe-derecha 1.5s 1;  
}
```

```
#zona-gestos.swipe-izquierda{  
  -webkit-animation: swipe-izquierda 1.5s 1;  
}
```

```
@-webkit-keyframes swipe-derecha{  
  0%,100% {  
    margin-left: 0px;  
  }  
  50%{  
    margin-left: 100%;  
  }  
}
```

```
@-webkit-keyframes swipe-izquierda{  
  0%,100% {  
    width: 100%;  
  }  
}
```

## PHONEGAP (I)

```
}  
50%{  
    width: 0%;  
}  
}
```

Y después añadimos el código para detectar el desplazamiento:

```
miHammer.on('swipe', function (ev) {  
    var clase;  
    var direccion = ev.direction;  
  
    if (direccion == 4) {  
        clase = "swipe-derecha";  
    }  
    if (direccion == 2) {  
        clase = "swipe-izquierda";  
    }  
  
    zona.className = clase;  
});
```

## ROTACIÓN

Igual que antes, creamos las animaciones:

```
#zona-gestos.rotate{  
    -webkit-animation: swipe-izquierda 1.5s 1;  
}
```

## PHONEGAP (I)

```
@-webkit-keyframes rotate{  
  100% {  
    -webkit-transform: rotate(360deg);  
  }  
  
}
```

Y añadimos el código. Atención que tenemos que habilitar la detección de rotación:

```
miHammer.get('rotate').set({enable: true});  
  
miHammer.on('rotate', function (ev) {  
  if (ev.distance>25){  
    zona.className = "rotate";  
  }  
});
```