

# IIC2026

# Visualización de Información

Alessio Bellino  
(2024 - 2 / Clase 19)

# Prototipado de InfoVis físicas con Protobject Code

- index.js
- physical-indicator.js
- physical.js
- speed.js
- speed2.js

+ Add File

Cada archivo JavaScript es asociado a un dispositivo

```
1 import Acceleration from './js/acceleration.js';
2 import Protoobject from './js/protoobject.js';
...t.js';
...tone@15.0.4/build/Tone.js";
...MIT, https://github.com/wouterbultman/
8 var KalmanFilter=function(){ "use strict";function s(t,i){for(v
9
10 var kf = new KalmanFilter({R: 1, Q: 3});
11
12
13 const p1 = {
14     harmonicity: 3,
15     modulationIndex: 10,
16     oscillator: {
17         type: 'sine'
18     },
19     envelope: {
20         attack: 0.1,
```

- index.js
- physical-indicator.js
- physical.js
- **speed.js**
- speed2.js

+ Add File

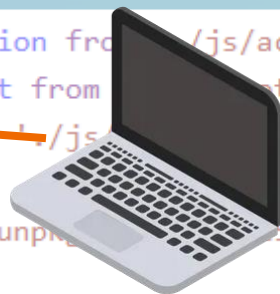
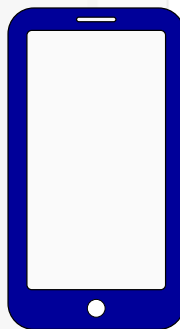
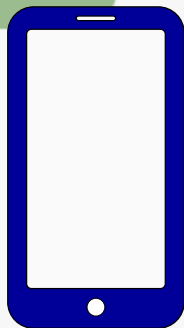
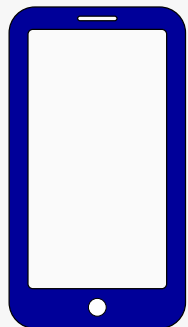
```

1 import Acceleration from './js/acceleration.js';
2 import Protoobject from './js/protoobject.js';
3 import Text from './js/text.js';
4
5 import "https://unpkg.com/tone@0.5.0.4/build/Tone.js";
6
7 /*kalm...ten, MIT, https://github.com/wouterbultman/tone.js
8 var K...ion(){"use strict";function s(t,i){for(v...
9
10 var...ilter({R: 1, Q: 3});
11
12 const p1 = {
13   harmonicity: 3,
14   modulationIndex: 10,
15   oscillator: {
16     type: 'sine'
17   },
18   envelope: {
19     attack: 0.1,
20

```

- index.js
- physical-indicator.js
- physical.js
- **speed.js**
- speed2.js

+ Add File



¡No solo smartphones!

```

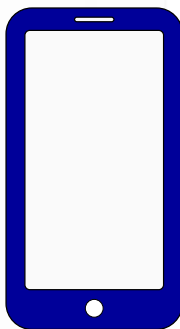
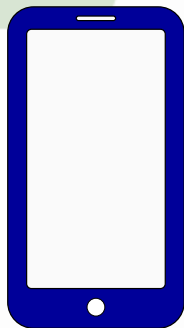
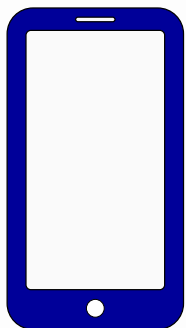
1 import Acceleration from './js/acceleration.js';
2 import Protobject from './js/protobject.js';
3 import Text from './js/text.js';
4
5 import "https://unpkg.com/tone.js@0.4/build/Tone.js";
6
7 /*kalm...ten, MIT, https://github.com/wouterbultman/tone.js
8 var K...tion(){"use strict":function s(t,i){for(v
9
10 var...alter({R...
11
12 const p1 = {
13   harmonicity: 3,
14   modulationIndex: 10,
15   oscillator: {
16     type: 'sine'
17   },
18   envelope: {
19     attack: 0.1,
20

```

- index.js
- physical-ind
- physical.js
- speed.js
- speed2.js

Los dispositivos  
pueden comunicar  
entre si a través de  
mensajes JSON

+ Add File

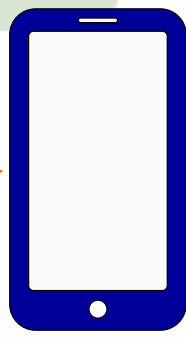
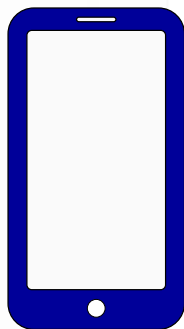


```

1 import Acceleration from './js/acceleration.js';
2 import Protobject from './js/protobject.js';
3 import Text from './js/text.js';
4
5 import "https://unpkg.com/tone@0.0.4/build/Tone.js";
6
7 /*kalm...ken, MIT, https://github.com/wouterbultman/tone.js
8 var K...tion(){"use strict";function s(t,i){for(v
9
10 var...alter({R: 1, Q: 3});
11
12 const p1 = {
13   harmonicity: 3,
14   modulationIndex: 10,
15   oscillator: {
16     type: 'sine'
17   },
18   envelope: {
19     attack: 0.1,
20

```

[Protobject Communication API](#)



The diagram illustrates the Protobuf Communication API. It features a background of JavaScript code snippets. On the left, a blue smartphone icon is connected by an orange curved line to a JSON file icon labeled 'json'. In the center, a tablet icon is connected by an orange curved line to a laptop icon. The code snippets include:

```
1 import Acceleration from './js/acceleration.js';
2 import Protobuffer from './js/protobuffer.js';
3 import Text from './js/text.js';
4
5 import "https://unpkg.com/protobuf@5.0.4/build/Tone.js";
6
7 /*kalm...ten, MIT, https://github.com/wouterbultman/...
8 var K...ion(){"use strict";function s(t,i){for(v...
9
10 var...alter({R: 1, Q: 3});
11
12 const p1 = {
13   harmonic: 3,
14   modulationIndex: 10,
15   oscillator: {
16     type: 'sine'
17   },
18   envelope: {
19     attack: 0.1,
```

## Protobject Communication API

# Comencemos con algo simple

Interruptor con luz

Switch

Lamp



# InfoVis físicas / no convencionales con Protobject Code

<https://youtu.be/rG7ijftXyRk>

# Contexto 1

Según el número de personas frente a la visualización, se muestran diferentes datos. Por ejemplo, cómo varía el consumo de energía eléctrica en función del número de personas que viven en un lugar.

[FaceSensor](#)

[Face detection guide](#) | [Google AI Edge](#) | [Google AI for Developers](#)

## Contexto 2

En un espacio cerrado con calefacción activa, la temperatura se mantiene estable gracias a un sistema de regulación. Sin embargo, si se abre una puerta, el calor acumulado comienza a escapar y el frío exterior a entrar, lo que provoca una **disminución en la temperatura**.

La velocidad de este descenso podría depender del grado de apertura: cuanto más abierta esté la puerta, más rápido bajará la temperatura. Si la puerta se abre solo un poco, la temperatura bajará más lentamente.

Orientation

# Interacción física

Al mover una casita para reproducir terremotos de distintas intensidades, se identificará cuál fue el terremoto real más cercano al movimiento generado.

Acceleration

# Interacción corporal

Los datos se filtran según la distancia a la visualización: cuanto más cerca estés, mayor será el zoom, y cuanto más lejos, menor. Además, puedes desplazarte en la visualización moviéndote hacia la derecha o izquierda.

# Interacción con gestos / física

Se visualizan distintas predicciones del precio del bitcoin — optimista, neutral y pesimista — basadas en

- gestos
- expresiones faciales
- interacción con objetos físicos

[HandSensor](#)

[FaceSensor](#)

[Acceleration](#)

[Hand landmarks detection guide](#)

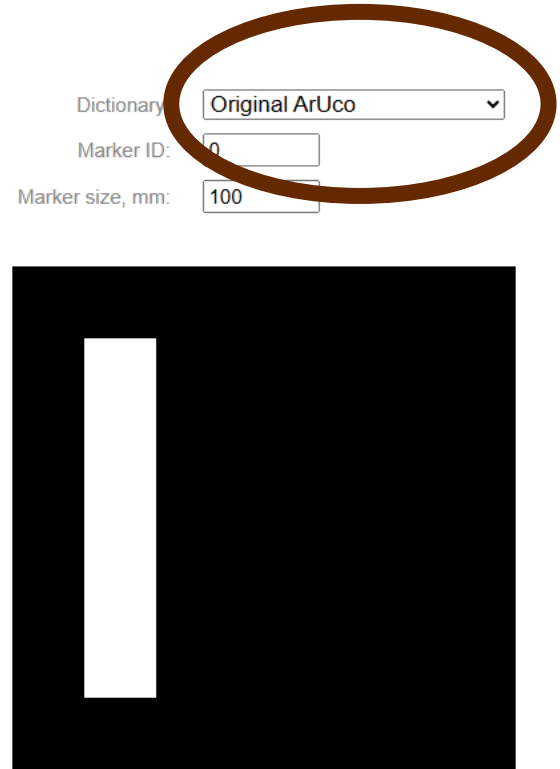
[Face landmark detection guide for Web](#)

# Interacción física (1)

Al agregar, retirar y/o reorganizar productos en el escenario (por ejemplo, porotos, té y atún), se controla la visualización de un gráfico que muestra las exportaciones de estos productos en varios países.

[Aruco](#)

[Online ArUco markers generator](#)



## Interacción física (2)

Al tocar un mapa físico, se escucha el monto en millones de dólares de las exportaciones de frutas del país seleccionado.

[HandSensor](#)

[TextToSpeech](#)

[Hand landmarks detection guide](#)



# IIC2026

# Visualización de Información

Alessio Bellino  
(2024 - 2 / Clase 19)