

# NUIact Documentation



Marc Sans Vera

25 de nov. de 2023

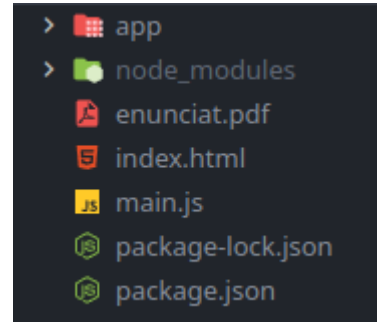
---

# Índex

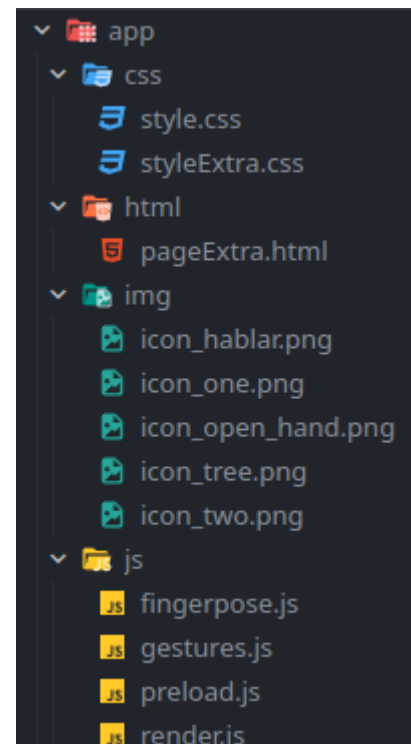
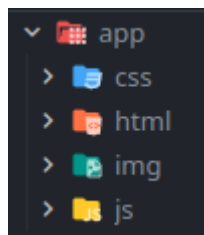
<b>Distribució del Projecte</b>	<b>3</b>
<b>Gestos i Comandaments Per Veu</b>	<b>4</b>
Implementació dels Gestos	4
Reconeixement de gestos	4
Implementació dels VoiceCommand	5
Reconeixement de Paraules Claus	6
<b>APP</b>	<b>7</b>
Pàgina Principal	7
Pagina Extra	7
<b>Bibliografia</b>	<b>8</b>

## Distribució del Projecte

Aquesta app s'ha desenrotllat principalment dins la carpeta app, encara que defora podem trobar els fitxers json que es generen quan cream el projecte electron, el main.js i l'index.html.



Dins app ens trobam 4 carpetes. Amb les quals separam les imatges, el css, l'html i el javascript



# Gestos i Comandaments Per Veu

## Implementació dels Gestos

Per poder implementar el reconeixement de gestos he emprat el projecte [fingerpose de andypotato](#). He separat el contingut de l'arxiu index.html en tres arxius diferents (css, html, js) i eliminat el contingut que trob innecessari, com les dues taules marcant la curvatura i direcció de cada dit o amagar els punts que es veuen en pantalla quan es detecta una mà.

## Reconeixement de gestos

Primer de tot he creat quatre gestos nous així com es pot veure en la documentació d'andypotato i els he inserit dins la llista "knownGestures" perquè es puguin detectar. Per acabar cridam una funció, "gestureActions", a la qual li passam el nom del gest que s'ha detectat.

```
const gestureActions = {  
  'one': () => {  
    alert("Open audio page")  
    window.appMessages.sendGestureResult("one");  
  },  
  'two': () => {  
    alert("Change theme")  
    window.appMessages.sendGestureResult("two");  
  },  
  'tree': () => {  
    alert("Close app")  
    window.appMessages.sendGestureResult("tree");  
  },  
  'hand': () => {  
    alert("devTools")  
    window.appMessages.sendGestureResult("hand");  
  },  
}
```

"gestureActions" és la funció que empram com a render per poder passar la informació al main.js a través del contextBridge amb la funció "sendGestureResult". Aquesta funció s'encarrega d'enviar un missatge amb una key, en aquest coincideix amb el nom del gest, però abans d'enviar aquest missatge bota un "alert" el qual avisa de l'acció que passarà.

El preload.js s'encarrega d'agafar aquesta key i enviar-la al main.js amb la funció "sendGestureResult".

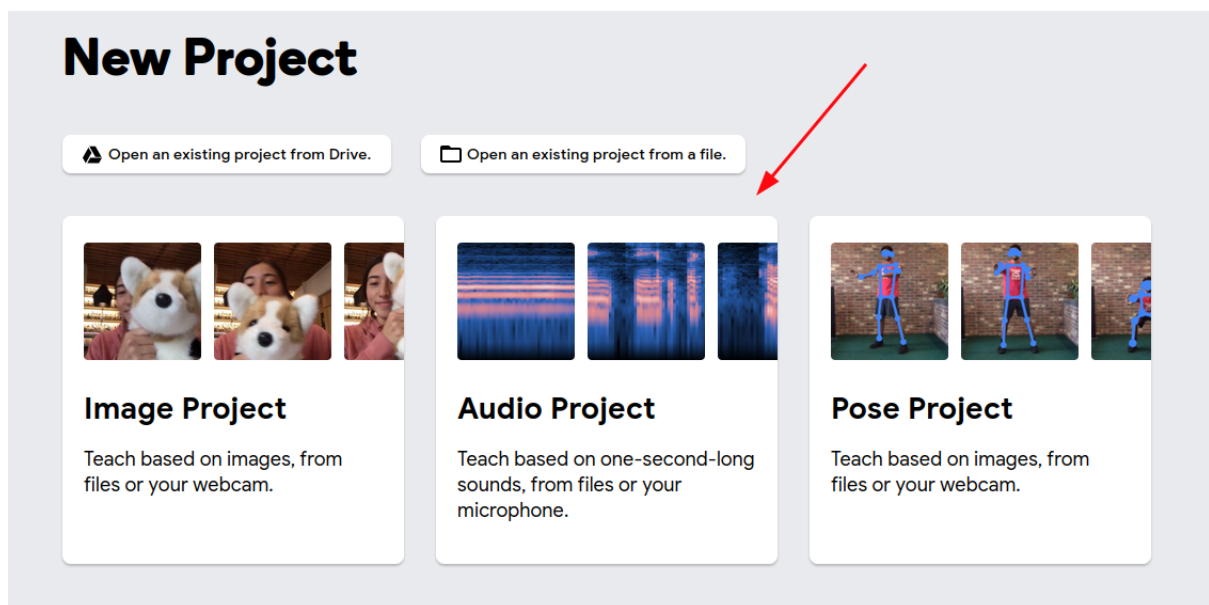
```
sendGestureResult: (gestureType) => {  
  ipcRenderer.send('sendGestureResult', gestureType);  
},
```

Al main.js agafam les dades i feim un switch on tenim un cases per cada gest amb una funció específica

```
ipcMain.on('sendGestureResult', (event, gestureType) => {
  switch (gestureType) {
    case 'one':
      win.loadFile('./app/html/pageExtra.html');
      break;
    case 'two':
      changeTheme();
      break;
    case 'tree':
      win.close();
      break;
    case 'hand':
      changeDevTools();
      break;
  }
});
```

## Implementació dels VoiceCommand

Per poder reconèixer paraules hem emprat la web [TeachableMachine](https://teachablemachine.withgoogle.com/). Amb aquesta creàvem un nou projecte d'àudio, cream les classes que ens interessin i unes mostres per cada classe.



Entrenam la intel·ligència artificial i pujam el projecte (no emprar firefox per pujar el projecte, no funciona).

## Reconeixement de Paraules Claus

Quan pujam el projecte es genera un codi JavaScript, copiam aquest i el modifiquem perquè en detectar una de les paraules es pugui emprar el contextBridge i fer que per cada paraula hi hagi una acció igual que feim amb els gestos. Una altra vegada empram l'alert per avisar de l'acció que passarà. També s'ha modificat la precisió que ha de tenir per detectar una paraula, per defecte està a 0.75, i se n'ha pujat a 0.80 perquè trobava que hi havia moltes vegades que detectava paraules claus fora que es diguessin, pujant-ho a 0.80 pareix que s'ha solucionat.

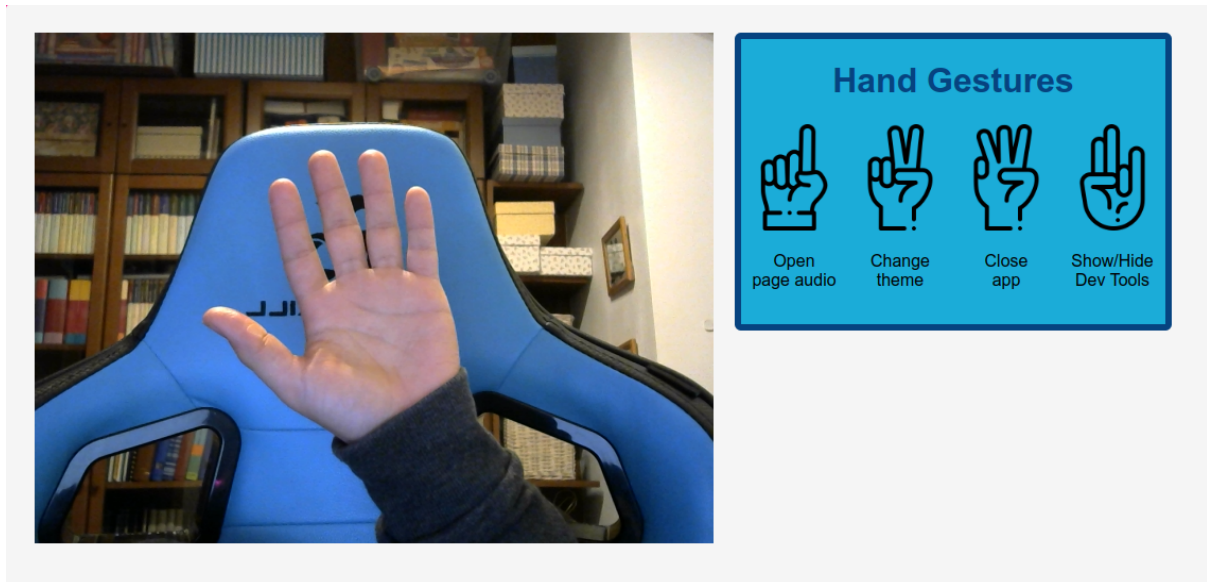
```
recognizer.listen(result => {
  const predictedWord = classLabels[result.scores.indexOf(Math.max(...result.scores))];
  labelContainer.innerHTML = predictedWord;

  if (predictedWord === "hola") {
    alert('Go to main')
    window.appMessages.sendVoiceResult('hola')
  }
  if (predictedWord === "adeu") {
    alert('Close app')
    window.appMessages.sendVoiceResult('adeu')
  }
  if (predictedWord === "tema") {
    alert('Change theme')
    window.appMessages.sendVoiceResult('tema')
  }
}, {
  includeSpectrogram: true,
  probabilityThreshold: 0.80,
  invokeCallbackOnNoiseAndUnknown: true,
  overlapFactor: 0.50
});
```

## APP

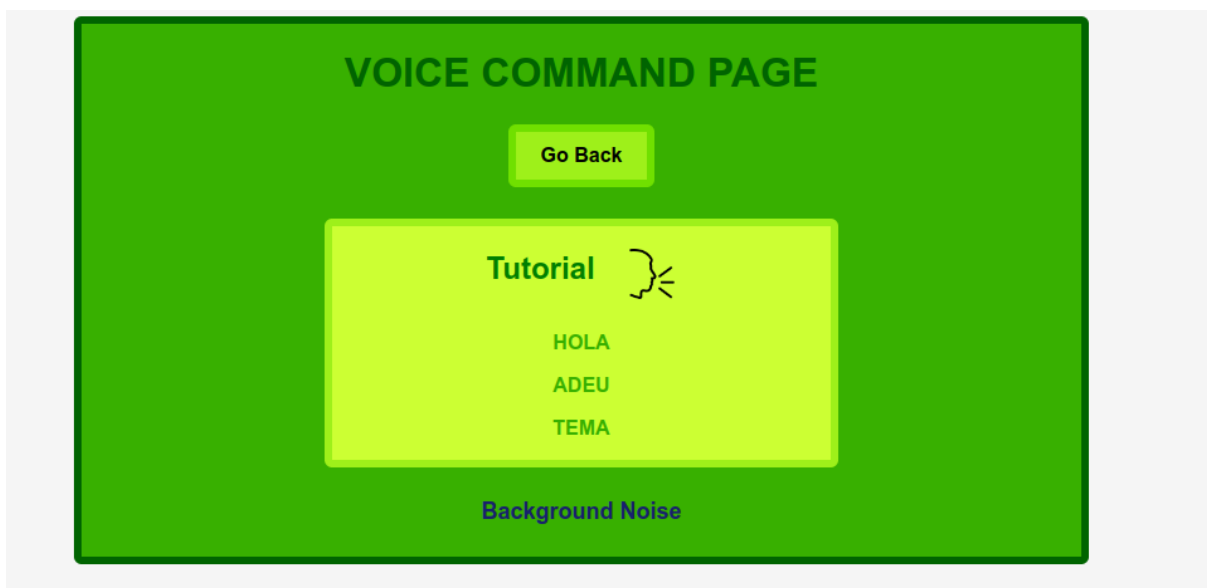
### Pàgina Principal

A la pàgina principal és on es detecten els gestos, mostrant la càmera i un petit tutorial indicant quins gestos ens trobam i quina acció realitzen.



### Pagina Extra

A la pàgina extra és on podem provar el voiceCommand, indicant quines són les paraules claus que es detecten, a més té un botó per tornar a la pàgina principal.



## Bibliografia

Electron: <https://www.electronjs.org/>

Paleta de colors: <https://coolors.co/palettes/trending>

Detectar veus: <https://teachablemachine.withgoogle.com/>

Detectar gestos: <https://github.com/andypotato/fingerpose>