Pau Casesnoves
Centre Integrat de Formació Professional

C/Joan Miró, 22 07300 Inca
Tel. 971 881711
secretaria@paucasesnovescifp.cat

NUI and ElectronJS

Course	23/24	**Group**	**S2P**	**Delivery date**	24/11 - 23:55
Module	Interfaces Development				
Title	Natural User Interfaces in Electron				

Type	Individual
Instructions	

Natural User Interfaces using Electron

Objective

The goal of this practice is to implement a natural user interface (NUI) using ElectronJS and understand the basic processes involved in defining and using classifiers to recognize gestures and voice commands.
The application will use gestures and voice commands to allow the user interaction.

General requirements

- Recognize 3 gestures and 3 voice commands(VC).
- Launch at least 3 different actions using gestures/VC.
- The main window will show indications of which gestures and words can be recognized by the application as well as their function.
- When a gesture/VC is recognized, a notification will be displayed to the user with information about the gesture and the action related.
- Use only the modules, libraries or code snippets needed by your application. Including code that never executes or libraries that are never used will be penalized.
- Do not show the menu at any window.
- Define your own modules, reutilize and organize your code.

Functional description

The application will initially display a window with information about the commands that can be recognized. A button will start the recognition system (RS).
This system must be tweaked by you from the examples reviewed at class.
You have to implement a controller over the RS that decides when to launch an action and when it can launch the next action (for example, stop the RS until the first action has completely finished).

Once a voice command or gesture is recognized, a notification will be displayed to the user with information about the function that will be executed next. When the notification closes the function will be launched.

Example of functions linked to gestures and voice commands

- open a different window for each recognized order
- maximize a window
- close a window
- modify the content or the CSS of some elements

Deliverables

- Your Electron project folder containing the necessary code to run it
- A document in PDF format explaining:
  - the application structure
  - the main concepts that involves voice command and gesture recognition used in your application
  - how your RS works, which actions are implemented
  - any issues you encountered during the practice and how did you solve it

Provide comments in your application code for clarity and organize your files in folders.

Qualification criteria

This activity corresponds to a 10% of the practical part.

Criterion 1: Application Setup and code organization (10%)

- 0 points: Application is not set up, or major errors exist.
- 1 point: The code is not organized at all.
- 2 points: Consoles show errors or some code is not used at all. Lack of comments.
- 3 points: Application is well-structured, the code is commented, organized and no errors were detected during testing.

Criterion 2: Gestures and Voice commands (30%)

- 0 points: Less than 3 gestures or VC are recognized and linked to different actions.
- 1 point: Just 3 gestures or VC are recognized but the actions are executed at the same renderer process.
- 2 points: The application recognizes 3 gestures and 3 VC that launches 3 different actions from the main process.
- 3 points: The application recognizes 3 gestures and 3 VC that launches more than 3 different actions from the main process.

Criterion 3: Recognition System (30%)

- 0 points: Not working.
- 1 point: It stops working during testing.
- 2 points: Works correctly but the code can be improved. There is no control over the RS.
- 3 points: Works correctly. The application controls over RS are correctly and efficiently implemented.

Criterion 4: Notifications (10%)

- 0 points: Not implemented.
- 1 point: Implemented but without gesture/command and action information.
- 2 points: Implemented using Window.alert()
- 3 points: Implemented using Electron API and Web Notifications (from the main and the renderer processes)

Criterion 5: Documentation (20%)

- 0 points: Not documented
- 1 point: Documentation has not been worked on enough.
- 2 points: Classifiers and RS are properly documented.
- 3 points: Classifiers, RS, the learning process and other external tools used are properly explained