

Memoria Técnica Leap

Práctica 1: Sensores

Nuevos Paradigmas de Interacción. Grupo 1.



**UNIVERSIDAD
DE GRANADA**

Silvia Barroso Moreno
Elena Merelo Molina
Adrián Rodríguez Montero
Carlota Valdivia Manzano

ÍNDICE

1. Introducción	3
2. Descripción de la aplicación	4
3. Diagramas	5
4. Estructura del proyecto	7
4.1. Clases empleadas	7
4.1.1 JFrames Forms	7
4.1.2 Sample.java	10
4.1.2.1 Función pCerrado	10
4.1.2.2 Función numDedosExt	10
3.1.2.3 Función giroMano	10
4.1.2.4 Función mover	11
4.1.2.5 Función aplaudir	11

1. Introducción

En esta memoria se describe nuestra aplicación de escritorio que les permitirá a todos los estudiantes de la Universidad acceder a amplios recursos. Se podrá obtener acceso a diversa información relativa a las distintas aulas, comedores, grados y los despachos de los profesores. En concreto nos hemos centrado en proporcionar la información necesaria a los estudiantes de la Escuela Técnica Superior de Ingeniería Informática y de Telecomunicaciones.

Esta aplicación utilizará la tecnología de Leap Motion para detectar determinados gestos y para permitirle a nuestra app ejecutar una funcionalidad u otra.

En los siguientes apartados se explicará toda la funcionalidad de nuestra app junto con las distintas clases de Java. Asimismo, se detalla la implementación de los gestos detectados por Leap Motion.

Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación



COMEDORES



GRADOS



DESPACHOS

2. Descripción de la aplicación

En primer lugar, cuando abramos la aplicación veremos una pantalla con cuatro botones. El primer botón nos llevará a información de las aulas, con el segundo obtendremos información de los comedores, con el tercero de los distintos grados y el último permitirá acceder a los despachos de los profesores. LeapMotion detectará cuantos dedos están activados y en función del número de dedos se seleccionará un botón u otro.

Las aulas se clasifican en aulas de teoría y prácticas, entonces al acceder a la información relativa a las aulas debemos de seleccionar si queremos información de aulas de teoría o prácticas. Esta selección se realizará con un gesto con Leap Motion y así conoceremos todas las aulas.

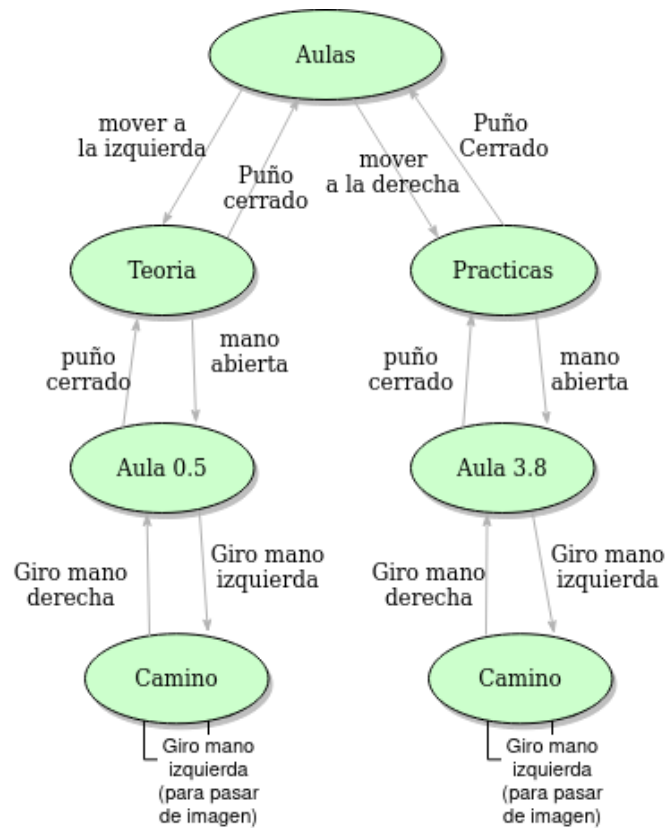
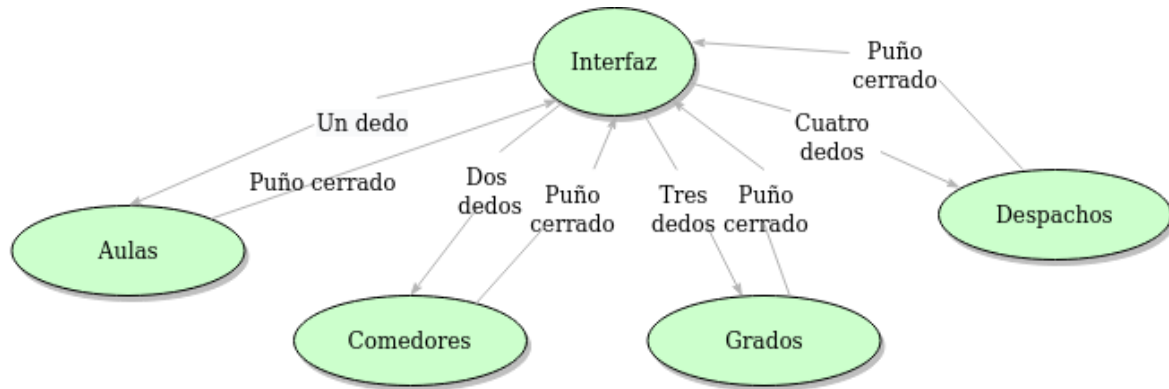
Por otro lado, si nos interesa saber un menú concreto de comedores, debemos de acceder a esta sección y seleccionar el día de la semana que nos interese. Además, podremos apuntarnos en una lista y la app registrará las personas que asisten a los comedores. No obstante también puedes desapuntarte si finalmente no puedes asistir.

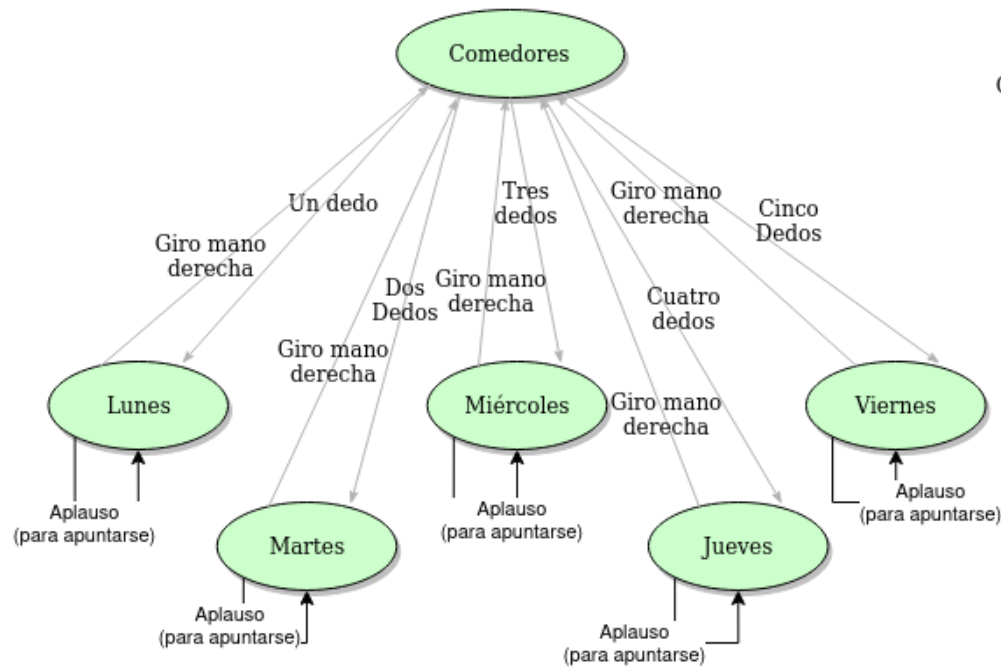
La tercera funcionalidad de nuestra app te permite visualizar los distintos grados de nuestra escuela y poder acceder a información relativa a cada uno de ellos.

Finalmente, la cuarta sección permitirá visualizar los despachos con objeto de poder ayudar a localizarlos, conocerlos y facilitar la comunicación profesor-alumno.

3. Diagramas

En los siguientes diagramas se muestran los gestos que se pueden realizar para ir cambiando entre las distintas vistas de la aplicación de escritorio.





4. Estructura del proyecto

En este apartado describiremos la estructura que se ha empleado para desarrollar la aplicación.

En primer lugar, la aplicación de escritorio se ha desarrollado mediante el software Apache Netbeans con el lenguaje de programación **Java**. Para su creación ha sido necesario definir dos paquetes: `com.company.interfaz` e `image`.

El primer paquete cuenta con todas las clases y **JFrames** necesarios para el funcionamiento de la aplicación. En él se han definido todas las vistas de la interfaz mediante distintos **JFrames forms** y una clase **Sample.java** en la cuál se han definido todos los gestos.

Para poder hacer uso del **Leap** hemos tenido que utilizar distintas librerías de Java destinadas al programa.

4.1. Clases empleadas

4.1.1 JFrames Forms

Para el desarrollo de la aplicación se han implementado 28 **JFrame forms**, estas se corresponden con las distintas vistas de la aplicación.

Cada uno de los **JFrames** contienen dos **JPanels** para tener de forma más estructurada la cabecera y el contenido. En cada **JPanel** se ha hecho uso de labels o botones, según los requisitos de cada vista.

En primer lugar hay una clase Interfaz que hereda de `javax.swing.JFrame`, en ella se muestra el menú principal de la aplicación. En ella hay cuatro botones que representan las distintas funcionalidades que aporta la aplicación.

El primero es un botón que te lleva a la clase **Aulas**, en ella puedes encontrar una nueva vista que te ofrece elegir entre aulas de teoría y de prácticas, siendo estos mismos botones que te llevan a dos nuevas vistas: **Teoría** y **Prácticas**.

La vista de **Teoría** está compuesta por tres botones en los cuales puedes obtener el camino a cada una de las distintas aulas de teoría. Del mismo modo, puedes ir a tres distintas clases de prácticas mediante la vista de **Prácticas**.

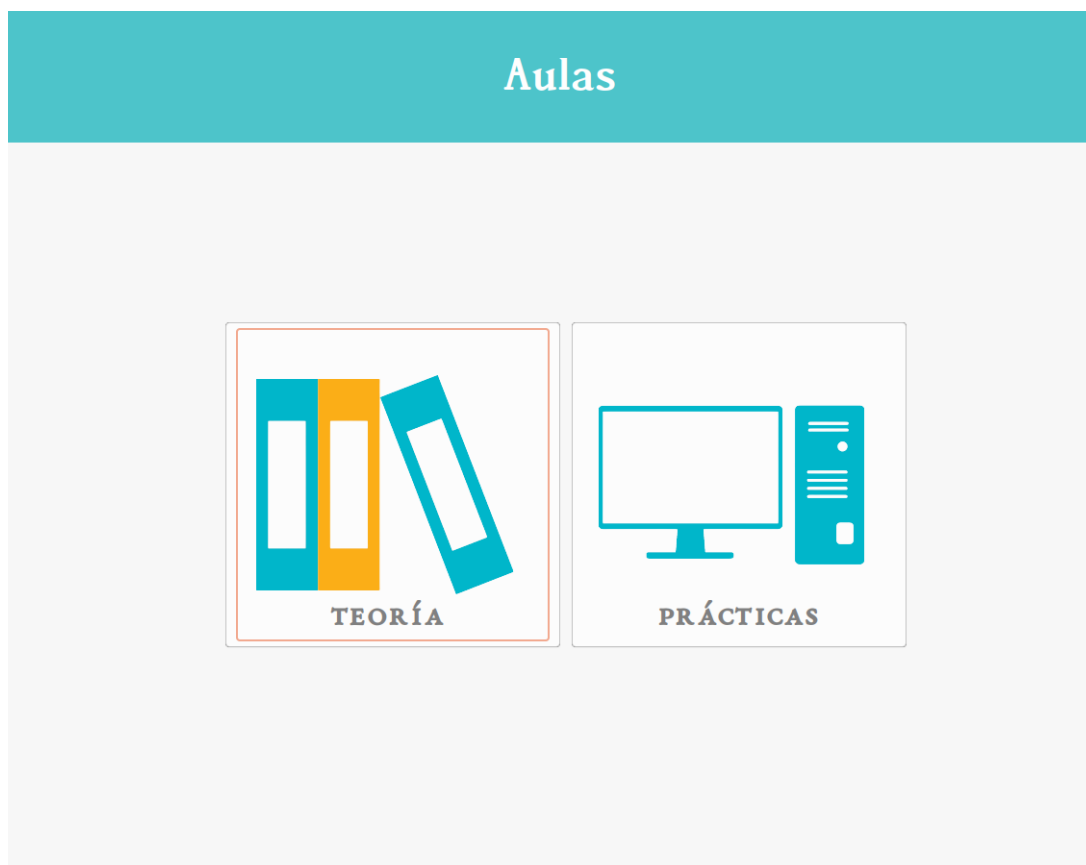
El segundo es un botón que te lleva a la clase **Comedores**, en ella se muestran cinco botones con los respectivos días de la semana. Al pulsar cada uno de ellos, la aplicación abre la vista correspondiente al día de la semana. En ella se muestra el menú del día en cuestión y dos botones para poder apuntarte y desapuntarte. Al seleccionar uno de ellos se mostrará un mensaje para cerciorar que la acción se ha realizado.

El tercer botón te lleva a la clase **Grados**, en ella se muestran cuatro botones con los respectivos grados: Ingeniería Informática, Ingeniería de Tecnologías de Telecomunicación, Doble Grado de Ingeniería Informática y Matemáticas, y Doble Grado de Ingeniería Informática y Administración y Dirección de Empresas.

Asimismo, cada uno de estos botones te lleva a distintos **JFrame forms**, con información sobre los distintos grados.

Por último, el cuarto botón te lleva a la clase **Despachos**. En esta última clase hay tres botones con distintos despachos. Al pulsar en cada uno de ellos, te lleva a una vista diferente donde te muestra mediante fotos cómo llegar al despacho en cuestión.

Todas estas transiciones entre botones se han implementado para poder realizar mediante gestos con **LeapMotion**. Dichos gestos los explicaremos con más profundidad a continuación.



jFrame de la vista Aulas

Menú Lunes

Primer Plato

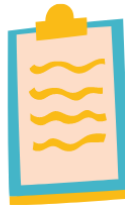
Arroz tres delicias
con salsa de soja
ó
Canelones vege-
tarianos

Segundo Plato

Albóndigas en salsa
de champiñones
ó
Cazón en adobo

Postre

Fruta ó lácteo



APUNTARME

DESAPUNTARME

JFrame de la vista Lunes

4.1.2 Sample.java

A continuación, vamos a mencionar los distintos gestos implementados para la tecnología LeapMotion. Todos estos gestos nos servirán para interactuar con nuestra aplicación.

4.1.2.1 Función pCerrado

Esta función me permite detectar si un puño está cerrado y cuando se detecte nuestra aplicación volverá a la pantalla anterior.

Primero, detecta una mano y una lista donde se almacenan los dedos de la mano que aparecen extendidos inicialmente. Vamos a estudiar el radio de la esfera de la mano a partir de **hand.sphereRadius()** y contar el número de dedos extendidos con **hand.fingers().extended().count()**. Si el radio de la esfera es menor que 40 y los dedos no se encuentran extendidos, entonces se detectará el gesto de **pCerrado**.

4.1.2.2 Función numDedosExt

En función del número de dedos extendidos nuestra aplicación accederá a las secciones de aulas, comedores, grados o despachos.

Si tenemos un dedo extendido se accederá a los distintos grados, si tenemos dos dedos se accederá a comedores, si introducimos tres dedos entonces nos situamos en la pantalla de grados y en el caso de cuatro dedos se accederá a los despachos.

El número de dedos extendidos se controlará con la funcionalidad **hand.fingers().extended().count()** que se comentó anteriormente.

3.1.2.3 Función giroMano

Este gesto nos permite visualizar las imágenes de las distintas aulas y despachos.

Un giro con la mano derecha nos permitirá mover una lista de imágenes desde la última imagen a la primera imagen, mientras que un giro con la mano izquierda nos permite desplazarnos de la primera imagen a la última. Por tanto, si por ejemplo tenemos 3 imágenes de un aula podemos visualizarlas de la primera a la última mediante tres giros con la mano izquierda y luego las podemos ver de la tercera a la primera con tres giros con la mano derecha.

Estos giros los hemos fijado a que sean mayores de 60° . Por otro lado, cabe resaltar que primero debemos de tener la mano apuntando en la dirección del eje Z ya que este ángulo se medirá en el plano formado por los ejes X y Z. Para determinar estas posiciones usaremos la normal de la mano y la dirección media de los dedos de la mano. Estos parámetros se detectarán en dos frames, el actual, esto es, la posición final de la mano, y el frame previo, la posición inicial de la mano.

4.1.2.4 Función mover

Este gesto consiste en un simple desplazamiento de la mano.

Si movemos la mano derecha hacia la izquierda se accede a las aulas de teoría y si movemos la mano izquierda hacia la derecha se accede a las aulas de prácticas. Análogamente, se volverá a usar la normal de la palma y su posición, y también se calculará la dirección media de los dedos de la mano.

Para detectar el movimiento se usarán dos frame el frame previo y el actual, en función de los parámetros comentados anteriormente en los distintos frames se detectará el gesto.

4.1.2.5 Función aplaudir

Este gesto nos permite aceptar un menú de comedores a partir de un aplauso.

Para determinar el aplauso detectaremos primeramente que en efecto estén ambas manos, luego que están abiertas, que las palmas pasan a estar paralelas y se mide la distancia entre las mismas. Por último, se considera que se ha aplaudido cuando acercamos las manos, con lo que la distancia mencionada es mayor que la nueva que se mide.