



TELECOMUNICACIÓN

Campus Sur
POLITÉCNICA

MEMORIA DE PRÁCTICA EXTERNA
ETS DE INGENIERÍA Y SISTEMAS DE
TELECOMUNICACIÓN
UPM

Elena Jurdao del Barrio



ÍNDICE DE CONTENIDOS

DATOS IDENTIFICATIVOS DE LAS PRÁCTICAS EXTERNAS:.....	1
1. INTRODUCCIÓN:.....	2
2. INFORMACIÓN DE LA ENTIDAD COLABORADORA	3
3. ENMARCAR LAS PRÁCTICAS EN EL CONTEXTO DE LA ENTIDAD	4
4. OBJETIVOS DE LAS PRÁCTICAS, TAREAS Y ACTIVIDADES REALIZADAS.....	5
5. TECNOLOGÍAS Y MEDIOS TÉCNICOS UTILIZADOS.....	7
6. COMPETENCIAS Y HABILIDADES ADQUIRIDAS CON LAS PRÁCTICAS	8
7. CONCLUSIONES	9
8. DIARIO DE PRÁCTICAS,.....	10

INSTRUCCIONES:

Se recomienda completar una cara por cada uno de los epígrafes a desarrollar, a excepción del nº4, "Objetivos de las prácticas...", que debe tener mayor contenido (dos caras)

1. INTRODUCCIÓN:

Las prácticas se han desarrollado en la empresa CITSEM (Centro de Investigación en Tecnologías Software y Sistemas Multimedia para la Sostenibilidad). En concreto, el trabajo realizado está englobado en el área de investigación: Grupo de Redes y Servicios de Próxima Generación (GRyS) que abarcan distintas temáticas: Tecnologías de Imagen y Vídeo, Redes y servicios y Tecnologías de software y servicios.

Las actividades llevadas a cabo en estas prácticas se han orientado en torno al reconocimiento de expresiones faciales de una persona a tiempo real, utilizando el dispositivo comercial, sensor KINECT v2 de Microsoft, se ha utilizado Visual Studio 2015 para poder realizar los códigos que controlaran el sensor con lenguaje C#.

El principal objetivo de las prácticas es realizar una interfaz natural, que permite la actuación del teclado mediante movimientos faciales. Con esto, se logra que una aplicación de ordenador que solicite el uso del teclado, pueda ser usada con movimientos faciales en tiempo real.

Por tanto se posibilita el acceso a aplicaciones interactivas desde el ordenador a personas que tienen alguna discapacidad motora o movilidad reducida.

Para conseguir esta finalidad se han realizado tareas con la cámara Microsoft Kinect v2 para comprensión de su funcionamiento y competencias principales. Así como del desarrollo de aplicaciones con Microsoft Visual Studio en concreto aplicaciones WPF.

2. INFORMACIÓN DE LA ENTIDAD COLABORADORA

La entidad donde se han llevado las prácticas ha sido CITSEM (Centro de Investigación en Tecnologías Software y Sistemas Multimedia para la Sostenibilidad). Este centro de investigación tiene como objetivo el aprovechamiento de las tecnologías básicas en las que sus miembros tienen experiencia, tecnologías software y multimedia, en un contexto de sostenibilidad.

Presenta cinco misiones principales, en las que se enfocarán todas las distintas áreas de trabajo que se llevan a cabo en dicha entidad. Estas misiones son:

- La realización por propia iniciativa de proyectos y otras tareas de investigación, desarrollo e innovación en el ámbito de las tecnologías software y multimedia con especial orientación hacia la sostenibilidad, así como la difusión de los resultados alcanzados con el fin de incrementar el conocimiento científico y los recursos tecnológicos al alcance de su entorno socio-económico.
- Colaborar con otras entidades (públicas o privadas) en la realización de proyectos de investigación, desarrollo y/o innovación en el ámbito de las tecnologías software y multimedia que contribuyan al logro de sus objetivos empresariales o de servicio público.
- Realizar trabajos específicos, informes o labores de asesoría para empresas privadas o instituciones públicas en el ámbito de las tecnologías software y multimedia.
- Impartir formación de postgrado tanto para salida profesional como investigadora (doctorado).
- Organizar e impartir cursos de especialización y perfeccionamiento, seminarios, conferencias y otras actividades de análoga naturaleza, en el ámbito de las tecnologías software y multimedia. Las áreas de trabajo en las que está dividido el CITSEM para el desarrollo de su investigación son: Grupo de Tecnología Software y Sistemas (SYST), Grupo de Diseño Electrónico y Microelectrónico (GDEM), y Grupo de Redes y Servicios de Próxima Generación (GRyS); el primero de la ETSISI y los otros dos de la ETSIST. En total cuenta con más de cuarenta investigadores a tiempo completo y se integra, como un actor más, en la estrategia de promover y fortalecer la I+D+i que se hace en el Campus Sur de la UPM.

3. ENMARCAR LAS PRÁCTICAS EN EL CONTEXTO DE LA ENTIDAD

La entidad donde se han desarrollado las prácticas, cubre tres áreas de investigación:

- Grupo de Tecnología Software y Sistemas (SYST).
- Grupo de Diseño Electrónico y Microelectrónico (GDEM).
- Grupo de Redes y Servicios de Próxima Generación (GRyS) que abarcan distintas temáticas: Tecnologías de Imagen y Vídeo, Redes y servicios y Tecnologías de software y servicios. En el presente caso, se ha trabajado en el área GRyS. En esta área se ha realizado un estudio y pruebas de reconocimiento facial utilizando dos de los sensores que posee el centro. El espacio de trabajo se ha situado en un laboratorio del centro de investigación CITSEM. Para la puesta en común con el resto del grupo de trabajo y tutora, se realizó en el mismo centro. Las personas con las que se han trabajado en dicha entidad han sido:

Martina Eckert: Doctora del CITSEM, tutora de prácticas.

4. OBJETIVOS DE LAS PRÁCTICAS, TAREAS Y ACTIVIDADES REALIZADAS

En las prácticas realizadas,, el objetivo principal se ha centrado en desarrollar una interfaz natural que permita la actuación del teclado mediante expresiones faciales recogidas mediante un sensor Kinect v2.

Para conseguir este objetivo principal, se han presentado una serie de propósitos a realizar:

- Estudio de reconocimiento de expresiones faciales
- Estudio de funcionamiento de Kinect v2
- Estudio del entorno de desarrollo SDK v2.0
- Estudio de las implementaciones realizadas de interfaz natural con movimientos corporales.
- Análisis del reconocimiento de expresiones faciales de Kinect.
- Estudio de lenguaje C#.
- Estudio de Windows Presentation Foundation (**WPF**) para crear modernas aplicaciones de escritorio empresariales en Windows.

En primer lugar, se procedió a realizar un estudio y revisión de los métodos que se han utilizado para poder hacer reconocimiento facial, esto se lo ha realizado gracias a la información previa generado por el grupo de investigación, la misma que se encuentra compartida en una carpeta de Moodle de la institución.

Revisando esta información lo más importante a destacar acerca del reconocimiento facial es la función face tracking proporcionada por el SDK v2.0 de Microsoft Kinect. EL SDK (Software Development Kit) incluye el conjunto de drivers, aplicaciones, herramientas y funciones que permiten a los desarrolladores programar aplicaciones utilizando como soporte su cámara. La función utilizada en este caso es face tracking que proporciona las herramientas necesarias para definir la malla que permite identificar los gestos faciales, gestionar esta información y representarla.

A continuación se desarrolla la función face tracking en profundidad:

Face tracking

El motor de reconocimiento de rostro Face tracking SDK analiza la entrada de una cámara Kinect, deduce la postura de la cabeza y las expresiones faciales y pone esa información a disposición de una aplicación en tiempo real. La función empleada en el desarrollo de la interfaz es una malla en 2D representada con 87 puntos como se indica en la siguiente imagen:

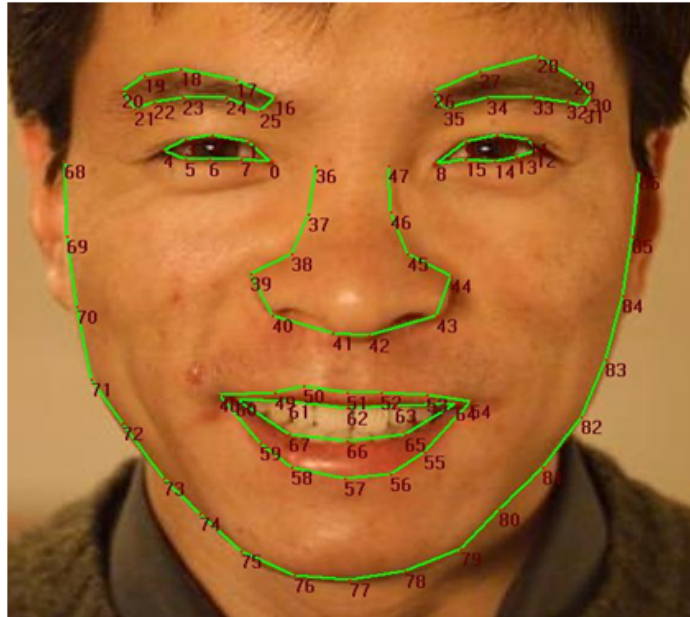


Figura 1.1: Puntos reconocidos por la función Face tracking para una malla en 2D

Estos puntos se devuelven en una matriz y se definen en el espacio de coordenadas de la imagen RGB devuelta desde el sensor Kinect. Existen 13 puntos adicionales (no mostrados en la figura) que incluyen: el centro del ojo, las esquinas de la boca y el centro de la nariz.

La función Face tracking es usada debido a que devuelve una serie de propiedades que son las utilizadas en la aplicación desarrollada. Estas propiedades son:

Propiedad	Descripción
Happy	El usuario parece estar sonriendo y está presentando una expresión feliz
Engaged	Determina si el usuario está comprometido con el contenido
WearingGlasses	El usuario lleva gafas
LeftEyeClosed	El ojo izquierdo del usuario está cerrado
RightEyeClosed	El ojo derecho del usuario está cerrado
MouthOpen	La boca del usuario está abierta
MouthMoved	La boca del usuario se ha movido
LookingAway	Determina si el usuario está mirando lejos del contenido.
Resultado de la detección	Descripción
Yes	Estamos muy seguros de que la propiedad es verdadera.
No	Estamos muy seguros de que la propiedad es falsa
Maybe	Estamos bastante seguros de que la propiedad es cierta.
Unknown	No tenemos suficiente información para hacer una determinación

Por tanto el diagrama de bloques de la interfaz desarrollada es el siguiente:

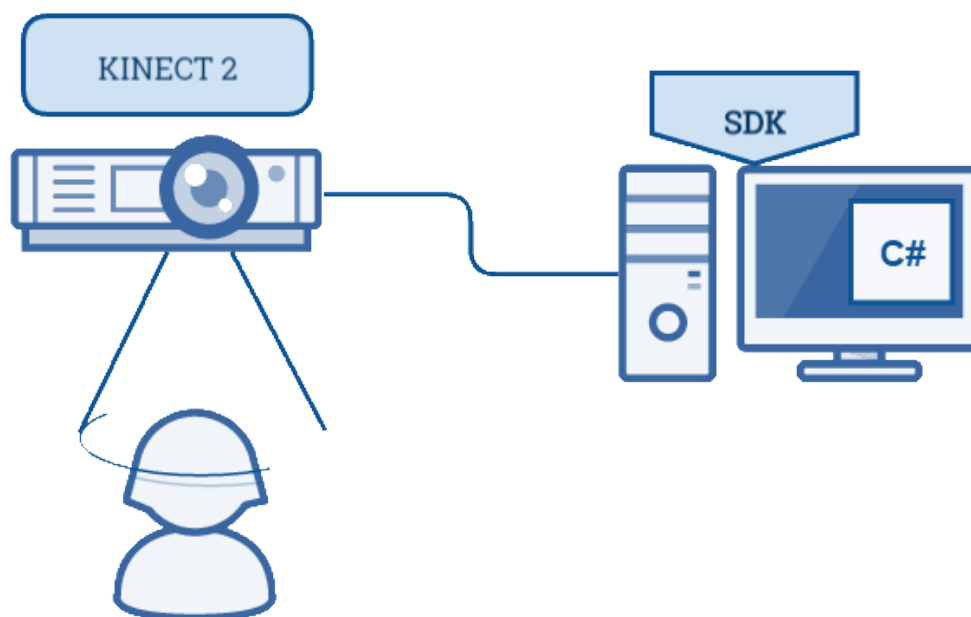


Figura 1.2: Diagrama de bloques de la interfaz implementada.

Donde en primer lugar la cámara Kinect 2 captura la cara del usuario. Después el SDK instalado en el ordenador permite a la cámara la comunicación con el ordenador y la realización del Face tracking. El último paso es usar el código desarrollado en C# para comunicarse con la aplicación activa, traduciendo las expresiones faciales en ordenes para esta.

Por el momento la aplicación desarrollada es capaz de comunicarse con una aplicación activa que solo utilice 4 comandos, es decir, que para su función solo sea necesario el uso de 4 teclas de ordenador. El ejemplo que se ha desarrollado ha sido el empleo de la interfaz natural para manejar la aplicación de Tetris, donde el ojo izquierdo se identifica con la tecla izquierda al igual que el ojo derecho, abrir la boca supone pulsar la tecla de flecha hacia arriba por lo tanto la pieza de tetris girará y sonreír determina pulsar la tecla de flecha hacia abajo por lo que la flecha del tetris bajará más rápido.

A continuación se proporcionan algunas imágenes como ejemplo de funcionamiento de la aplicación:

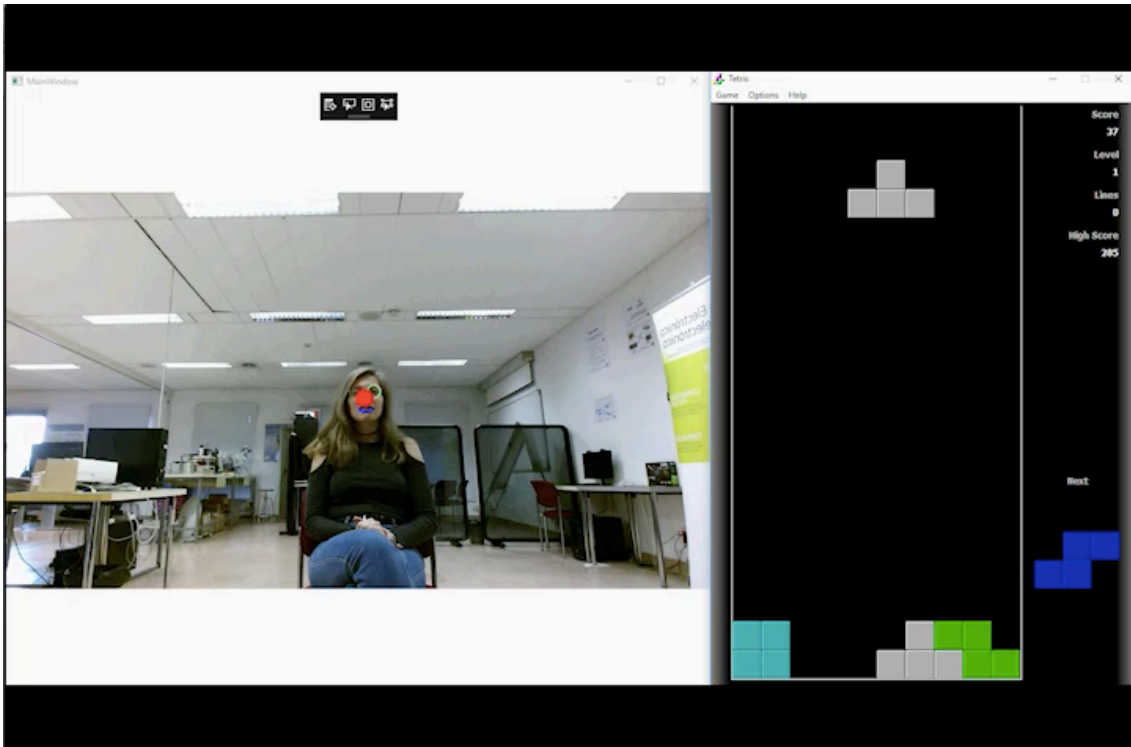


Figura 1.3: Ojo izquierdo cerrado representado por la desaparición de la esfera verde lo que supone que la pieza se mueve a la izquierda

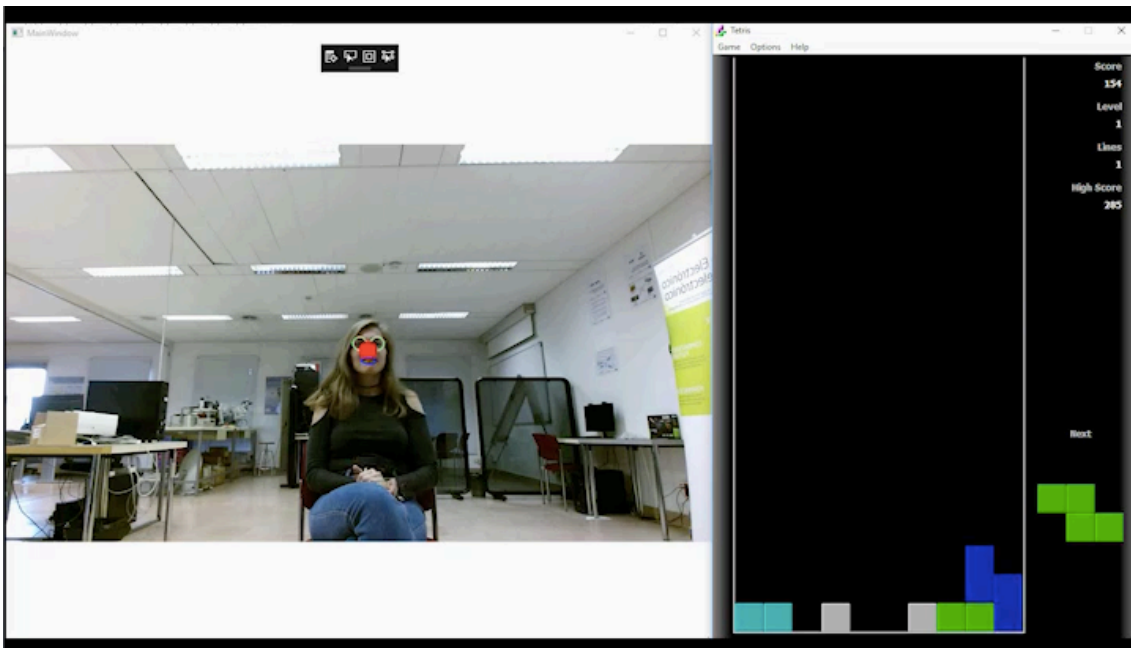
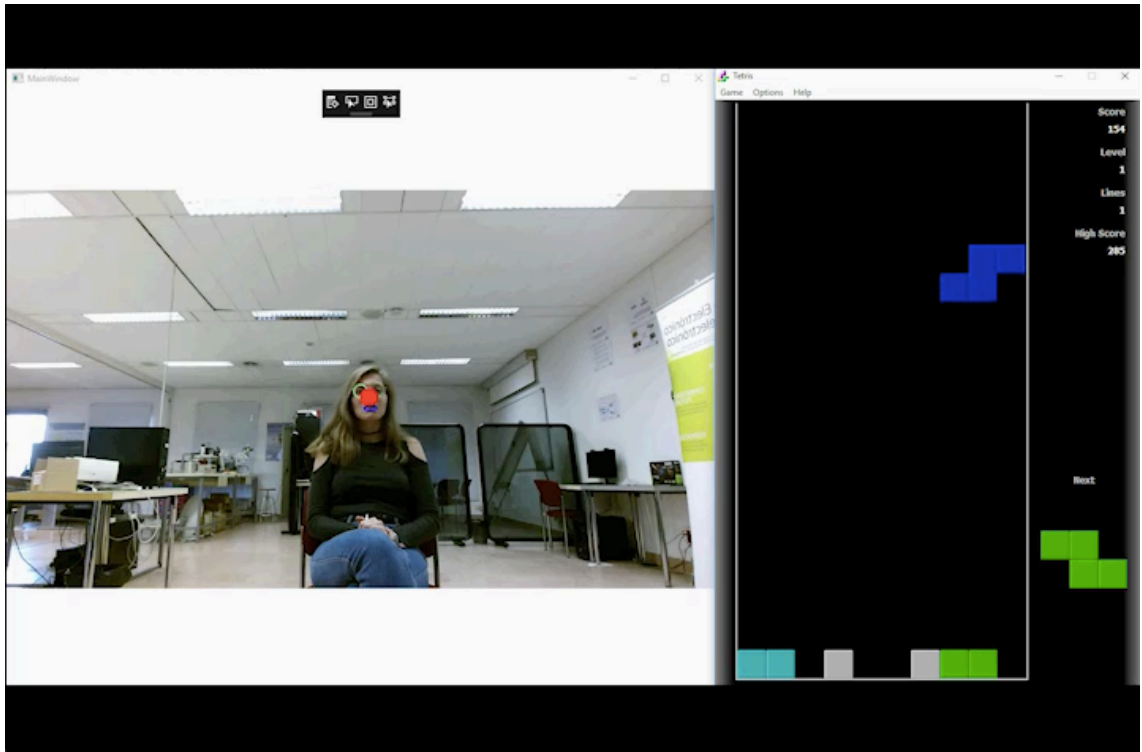


Figura 1.4: El usuario ha sonreído luego la pieza se ha desplazado rápidamente hacia abajo.



5.

FIGURA 1.5: OJO DERECHO CERRADO REPRESENTADO POR LA DESAPARICIÓN DE LA ESFERA VERDE LO QUE SUPONE QUE LA PIEZA SE MUEVE A LA DERECHA

TECNOLOGÍAS Y MEDIOS TÉCNICOS UTILIZADOS

- Ordenador con Windows 10.
- Entorno de desarrollo SDK v2.0
- Kinect versión 2
- Adaptador de Kinect versión 2 para PC
- Microsoft Visual Studio 2015, para poder desarrollar el código necesario C#.
- Bases de datos científicas para obtener la información correspondiente al estado del arte de las tecnologías concretas.

6. COMPETENCIAS Y HABILIDADES ADQUIRIDAS CON LAS PRÁCTICAS

Las competencias adquiridas en las prácticas realizadas son:

- Conocimiento del lenguaje de programación C# para desarrollo de aplicaciones basados en arquitecturas tipo Microsoft.
- Conocimiento de sensores de reconocimiento de video y tratamiento digital de procesado de imágenes.
- Conocimientos avanzados del área de Ingeniería de Telecomunicación con capacidad para aplicarlos de una forma profesional y para innovar y desarrollar elementos y sistemas en ese área.
- Capacidad de búsqueda y selección de información, de razonamiento crítico y de elaboración y defensa de argumentos dentro del área.
- Capacidad de abstracción, de análisis y de síntesis y de resolución de problemas.
- Capacidad para aprender de manera autónoma nuevos conocimientos y técnicas adecuados para la concepción, el desarrollo o la explotación de sistemas y servicios de telecomunicación.
- Capacidad de utilizar la programación orientada a objetos para el desarrollo de aplicaciones que involucren procesado digital de imagen.

Las asignaturas impartidas en el grado relacionadas con el trabajo que se ha llevado a cabo son:

- Programación I
- Programación II

7. CONCLUSIONES

- Se ha logrado entender a rasgos iniciales el lenguaje de programación C#, el cual, al ser un lenguaje orientado a objetos está muy relacionado con instancias de Java y C++ en su estilo de programación. Por tanto, las librerías a utilizar en Kinect requieren de un manejo de objetos para cada trama (imagen) que se procese en tiempo real.
- Kinect v2, al ser una versión mejorada de su predecesora dispone de sensores con mayor sensibilidad que permite un tratamiento digital de imágenes en HD con mucha más información.
- Las librerías de SDK 2.0 al mejorar los sensores de la cámara , han mejorado de la misma forma lo que ha facilitado el tratamiento de los datos del reconocimiento facial y por tanto la comunicación con la aplicación activa para el desarrollo de la interfaz natural.
- Se deja una línea abierta hacia la mejora de la aplicación , incluso el desarrollo de esta para una malla 3D que supondrá una mejor precisión del reconocimiento facial y menos limitación a la hora de identificar expresiones con comandos.
- La creación de una interfaz destinada a personas con movilidad reducida ha sido realmente gratificante ya que aproximadamente 250.000 personas solo en España tienen que lidiar con ello.
- Lamentablemente al tratarse de un producto de Microsoft, la información difundida es muy poca, lo cual exige al desarrollador a formar grupos de estudio o investigación, lo cual, ha permitido llegar a información que ha sido creada por desarrolladores de Microsoft para obtener explicaciones de librerías y métodos.
- La labor del tutor y su función desempeñada en estas prácticas, ha sido esencial en esta investigación. Su continuo interés, tanto a la hora del estudio de las herramientas y conocimientos a adquirir, como a la hora de realizar las implementaciones posteriores.

DIARIO DE PRÁCTICAS

Semana	Actividad
12-18 septiembre	Inicio de prácticas, explicación del grupo de estudio
19-25 septiembre	Reunión con previos desarrolladores de aplicaciones para Kinect v2 y v1. Análisis y entendimiento de Kinect v1
26 septiembre-2 octubre	Incursión en Kinect v2.
3-9 octubre	Análisis y entendimiento del SDK 2.0
10-16 octubre	Estudio y aprendizaje de C# y revisión de documentación.
17-23 octubre	Estudio y aprendizaje de C#.
24-30 octubre	Estudio y aprendizaje de la creación de aplicaciones WPF con la creación de algún ejemplo
31 octubre-6 noviembre	Estudio y aprendizaje de la función Facetracking con la creación de algún ejemplo.
7-13 noviembre	Desarrollo de código de la interfaz.
14-20 noviembre	Desarrollo de código de la interfaz.
21-27 noviembre	Desarrollo de código de la interfaz.
28 noviembre- 4 diciembre	Desarrollo de código de la interfaz.
5-11 diciembre	Realización de pruebas y mejoras del código
12-20 diciembre	Optimización del código y los recursos, generación de la documentación requerida.