



ESCUELA SUPERIOR POLITECNICA DEL LITORAL

FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN

**“SISTEMA MULTIMEDIA INTERACTIVO DE BÚSQUEDA DE
INFORMACIÓN POR MEDIO DE UNA PANTALLA
MULTITOUCH DE BAJO COSTO”**

INFORME DE PROYECTO DE GRADUACIÓN

Previo a la obtención del título de:

**INGENIERO EN CIENCIAS COMPUTACIONALES
ESPECIALIZACIÓN SISTEMAS MULTIMEDIA**

Presentado por:

Andrés Norberto Vargas González

GUAYAQUIL – ECUADOR

2010

AGRADECIMIENTO

Primero que nada agradezco a Dios por permitirme disfrutar de la familia que tengo, así mismo les agradezco a ellos que siempre estuvieron junto a mí en todo momento, sin duda alguna todo lo que soy y puedo llegar a ser se los debo a ellos.

Un agradecimiento especial a todos mis profesores en especial a la Msc. Carmen Vaca en mi vida estudiantil, y a la Dra. Katherine Chiluiza por el tiempo y esfuerzo de ser mi Directora de Tesis y la confianza puesta en el proyecto.

DEDICATORIA

Le dedico este trabajo a Dios por ser la luz que me ilumina siempre.

A mi familia que siempre me ha apoyado incondicionalmente en todas mis decisiones.

A las personas que de una u otra forma me ayudaron a lo largo de toda mi carrera Universitaria.

TRIBUNAL DE SUSTENTACION

Ph. D.Katherine Chiluita

DIRECTORA DE PROYECTO DE GRADO

Ph. D.Sixto García

PRIMER VOCAL PRINCIPAL

Ph. D.Xavier Ochoa

SEGUNDO VOCAL PRINCIPAL

DECLARACIÓN EXPRESA

“La responsabilidad del contenido de este informe de proyecto de graduación me corresponde exclusivamente; y el patrimonio intelectual de la misma a la Escuela Superior Politécnica del Litoral”.

(Reglamento de exámenes y títulos profesionales de la ESPOL)

Andrés Vargas González

RESUMEN

En el presente trabajo se desarrolló un Sistema Multimedia para ser usado sobre una Pantalla Multitouch de bajo costo, la pantalla se construyó con el fin de analizar otras formas de interacción, y el sistema se realizó con el objetivo de medir la interacción de un usuario sobre un dispositivo multitouch.

Para el desarrollo de este producto se revisaron los conceptos de protocolos de intercambio de información y visión por computador para la Pantalla Multitouch; y frameworks de desarrollo para la aplicación que se ejecuta sobre la Pantalla.

También, para el desarrollo de la aplicación se analizaron diferentes tipos de librerías y procedimientos idóneos, así como también se hizo un análisis profundo de cada uno de los componentes de la pantalla.

Luego de la revisión del Estado de Arte se definió la arquitectura y Diseño del Sistema. Se comenzó describiendo el proceso de construcción de la Pantalla Multitouch por prototipos, donde se destaca el uso de CommunityCoreVision para el seguimiento de dedos.

Luego se describió los módulos de la aplicación a ser ejecutada, uno por uno, describiendo claramente lo que el módulo recibe, procesa y devuelve. Las herramientas usadas se las describió en la sección de Implementación. Éstas fueron PyMT como framework de desarrollo para manejar y presentar datos y SQLAlchemy para el manejo asíncrono de datos desde la Base.

Al final de la implementación del proyecto se procedió a realizar pruebas con los usuarios, donde se observó una aceptación de estos con respecto a la manipulación de este tipo de interfaces. No se presentaron mayores problemas en la ejecución de las tareas expuestas a ser evaluadas.

INDICE GENERAL

AGRADECIMIENTO	ii
DEDICATORIA	iii
Capitulo 1	3
1. Análisis del Problema	3
1.1 Antecedentes	3
1.2 Descripción del problema	6
1.3 Justificación.....	8
1.4 Propuesta de solución.....	9
Capitulo 2	11
2. Revisión del estado del arte	11
2.1 Protocolo TUIO.....	11
2.2 Descripción	12
2.3 Especificaciones.....	14
2.4 OPEN SOUND CONTROL.....	15
2.4.1 Descripción.....	16
2.4.2 Formato	16
2.4.3 Implementaciones.....	18
2.5 TUIOTracker Implementations	19
2.5.1 Reactivision	19
2.5.2 Touchlib.....	22
2.5.3 CCV	23
2.6 Análisis comparativo de librerías disponibles.....	26
Capitulo 3	30
3. Análisis y diseño de la aplicación.....	30
3.1 Análisis y diseño de la Pantalla.....	44
3.2 Módulo de búsqueda por tags	59
3.3 Ubicación de trabajadores	60
3.4 Proyectos de investigación realizados.....	61
3.5 Relaciones de dependencia entre Personas y Proyectos	62
3.6 Diseño del Plan de pruebas	64
Capitulo 4.....	66

4. Implementación	66
Capitulo 5	76
5. Pruebas y análisis de resultados.....	76
5.1 Pruebas de funcionalidad	77
5.2 Pruebas de usabilidad.....	81
5.3 Análisis de resultados.....	83

CONCLUSIONES

RECOMENDACIONES

BIBLIOGRAFÍA

ANEXOS

ANEXO A.- Descripción de cada una de las Tablas de la Base de Datos

ANEXO B.- Prueba de Usabilidad

INDICE DE TABLAS

Tabla 1. Tipos semánticos de mensajes set en un objeto TUIO.....	15
Tabla 2. Análisis de las características de los Trackers.....	26
Tabla 3. Lista de Materiales Primer Prototipo	45
Tabla 4. Lista de Dispositivos del prototipo 2 y Medidas	47
Tabla 5. Lista de Materiales del Segundo Prototipo	51
Tabla 6. Lista de Materiales y Costo del último Prototipo	54

INDICE DE FIGURAS

Figura 1. Formato mensaje OSC	17
Figura 2. Formato de un mensaje XML.....	17
Figura 3. Reconocimiento de Patrones	20
Figura 4. Archivos de Configuración, e interfaz Reactivision.....	21
Figura 5. Interfaz de Touchlib.....	23
Figura 6. Interfaz de CCV.....	25
Figura 7. Administración Web Movid	29
Figura 8. Presets personalizables según necesidad de programador ...	29
Figura 9. Gesto de Arrastrar y Soltar	31
Figura 10. Gesto de Rotar	32
Figura 11. Gesto de Escalar	33
Figura 12. Teclado QWERTY Táctil.....	34
Figura 13. Palabras Claves	35
Figura 14. Detalles de Personas	36
Figura 15. Proyectos y Relación de Personas.....	37
Figura 16. Esquema de Funcionamiento de la Pantalla Multitouch	39
Figura 17. Arquitectura del Sistema	40
Figura 18. Selección de palabras claves.....	42
Figura 19. Diseño de la Interfaz.....	43
Figura 20. Primer Prototipo de la Pantalla Multitouch	45
Figura 22. Vista lateral de la mesa	48
Figura 24. Prueba de precisión Segundo Prototipo	52
Figura 25. Segundo Prototipo.....	52
Figura 27. PS3EYE.	56
Figura 28. Lámparas de Luz Infrarroja.....	57
Figura 29. Esquema de las lámparas de Luz Infrarroja.....	58
Figura 30. Esquema E/S del Módulo de Búsqueda por Tags.....	59
Figura 31. Esquema E/S del Módulo de Ubicación de Trabajadores....	60
Figura 32. Mapa de la ESPOL	61
Figura 33. Esquema E/S del Módulo de Proyectos de Investigación....	62
Figura 34. Diseño Estructural de la Base de Datos	63
Figura 35. Esquema general del Proyecto.....	67
Figura 37. Posición del Proyector, Espejo y Cámara.....	68
Figura 38. Vista interior de la Mesa Multitouch	69

Figura 39. Medidas de los elementos internos de la mesa multitouch	70
Figura 40. Vista exterior de la Mesa Multitouch.....	71
Figura 41. Gráfico de la Funcionalidad del Sistema respecto a la correcta realización de las tareas por parte de los usuarios.	78
Figura 42. Gráfico del tiempo promedio que requirió cada usuario en completar cada tarea.	80
Figura 43. Gráfico de los promedios de dificultad por tarea.	81
Figura 44. Gráfico que muestra promedio de dificultad por tareas y general	82

INTRODUCCIÓN

Actualmente, ESPOL no posee un Sistema, que ayude a encontrar una persona en el Campus, existe un dispositivo instalado en rectorado que sirve para ubicar las facultades en un mapa interactivo que es touch, pero como tal no tiene la funcionalidad de encontrar personas ni proyectos asociados a aquellas personas.

Por otro lado, la evolución de la tecnología y ambientes gráficos, ha provocado nuevas formas de interacción entre humanos y computadores, ahora la forma de interacción está tendiendo al uso de dispositivos multitouch, que por medio de recursos multimedia como videos, imágenes y sonidos generan un ambiente más natural al usuario.

La utilización de este dispositivo significaría un gran paso al lograr involucrar a cualquier persona en el uso de sistemas computacionales, por medio de una interacción directa.

El uso de aplicaciones de visión por computador y procesamiento digital de imágenes nos brindan la posibilidad de crear ambientes inmersivos al

usuario, de tal forma que, se busca acercar más los ambientes virtuales a la realidad. Lo que se espera es que los usuarios intuitivamente sepan cuál es el siguiente paso a realizar en la aplicación, de tal forma que el usuario naturalmente la manipule. Cada uno de los objetos de la aplicación posee características que se asemejan a la realidad, como agrandar, encoger y mover.

El principal objetivo de esta tesis es que las personas se acerquen a la pantalla y se sientan motivadas al uso de la aplicación, y que cada acción que realicen sobre la superficie, sea realizada de una forma natural.

El uso de las manos como medio de interacción no es algo nuevo y ya tiene su historia dentro de las interfaces táctiles, así como de ambientes virtuales.

CAPITULO 1

1. ANÁLISIS DEL PROBLEMA

En este capítulo se describe el problema que se espera resolver, la justificación para implementarlo y posible ubicación de la solución planteada en el presente trabajo en el Campus de la ESPOL.

1.1 ANTECEDENTES

El uso de pantallas táctiles no es un tema nuevo en el ámbito de aplicaciones computacionales, algunas alternativas se han desarrollado, a nivel de hardware, para su elaboración. Sin embargo, en los últimos 5 años se han implementado soluciones a nivel de software, que incorporan uso de visión por computadora y elementos físicos para lograr construir dispositivos de este tipo, con la ventaja de no contar con ningún tipo de circuitería.

Los inicios de proyectos en los que se usaron pantallas táctiles, empezaron a implementarse desde la década de los 70. En el año

1971, el primer sensor táctil fue desarrollado por el Doctor Sam Hurst (fundador de Elographics), mientras era instructor en la Universidad de Kentucky. Este sensor no era transparente como las pantallas táctiles actuales pero sin duda alguna fue un gran paso en la evolución de la tecnología de las pantallas táctiles.

Tres años después Elographics desarrolló una pantalla táctil incorporando una superficie transparente. Esta empresa continuó con la evolución de su tecnología y años después en 1994 pasó a llamarse EloTouchSystems [10].

Si bien este tipo de tecnología tiene mucho tiempo de desarrollo, no es sino, hasta hace unos 5 años en los que se ha popularizado por el uso de las pantallas táctiles en dispositivos móviles. Su interacción ha mejorado debido a que los usuarios utilizan sus móviles. Dispositivos actuales como el iPhone [12] y algunos teléfonos Android [13] poseen incluso pantallas multitouch. Las pantallas multitouch son superficies que soportan la directa interacción del usuario sobre ellas, pudiendo interactuar varios usuarios a la vez, uno de estos dispositivos es el Microsoft

Surface[21]. En este trabajo nos referiremos con el nombre de multitoucha este tipo de pantallas.

Las pantallas multitouch han logrado provocar un ambiente más natural que una interfaz por teclado o mouse, debido a que los usuarios pueden usar sus manos, específicamente dedos, como un dispositivo de entrada de la pantalla. Entonces es aquí donde podemos definir un cambio generacional en la forma de interacción con las computadoras, a esta nueva generación se la ha denominado (Natural User Interfaces o Interfaces Naturales de Usuario) NUI.

Las NUI no solo proveen de una directa manipulación al contenido, sino que además permiten a múltiples usuarios trabajar sobre la superficie generando un ambiente colaborativo por medio de una interfaz intuitiva, dejando de lado el uso del mouse y del teclado como medios de entrada.

En capítulos siguientes se analiza las ventajas y desventajas que puede tener la interacción con dispositivos como los mencionados

previamente. Además se prevé al finalizarla dejar planteado el inicio de posibles investigaciones en el uso de pantallas multitouch para proyectos de Información.

1.2 DESCRIPCION DEL PROBLEMA

Generalmente, cuando una persona ingresa a un edificio, campus universitario o instalaciones de una institución que desconoce, en busca de información de una persona, enfrenta algunas dificultades debido a ese desconocimiento. Incluso debe esperar turnos para lograr conseguir un propósito particular, la percepción en relación al servicio de información de la Institución puede verse afectada por el tiempo de respuesta de la persona que atiende; además, de la calidad y utilidad de la información dada. Otra evidencia de este problema es que en la mayoría de Instituciones, por lo general, la consulta de información se la realiza a una persona, quien no siempre cuenta con el suficiente conocimiento para proveer información, o bien el tiempo que se toma en realizar la búsqueda supera lo que una persona comúnmente prefiere esperar.

Específicamente, en ESPOL que es un campus universitario de 720 ha, las personas que por primera vez visitan la universidad, tienden a perderse y buscar ayuda en el edificio más cercano que encuentren. En el caso en que se busque a una persona dentro del mismo campus, si no se conoce la ubicación exacta de su oficina, el visitante dedicará algún tiempo hasta encontrar la persona con la que quiere entrevistarse; lo mismo sucede si el visitante está interesado en un proyecto conocido o en un centro de investigación o de desarrollo, al cual quiere visitar o contactar. Actualmente se dispone de un equipo en Rectorado, que es poco visible, no provee de información sobre personas, proyectos, sino solo ubicación de edificios o centros y no es tan sencillo de usar.

En el caso del Parque del Conocimiento (PARCON), donde coexistirán más de 5 centros de Investigación, y en cada uno trabajarán cientos de personas manejando proyectos en las diferentes ramas de la ciencia, el problema de encontrar personas, proyectos o centros, crecerá y se hará más evidente.

1.3 JUSTIFICACION

Este proyecto busca proveer a los usuarios, de un Sistema basado en la directa interacción sobre un dispositivo de entrada y salida construido en ESPOL que le permitirá realizar búsquedas de personas, proyectos, entre otros, de manera sencilla y natural. Este dispositivo, constará de una mesa interactiva multitouch, de bajo costo, y que permita interacción con las manos de varios usuarios trabajando simultáneamente, basado en una serie de reglas de inteligencia artificial. La información que provea el usuario al sistema puede ser gráfica o textual y consecuentemente la información que se devolverá será: nombres de personas encontradas, fotografías, proyectos asociados, lugar de ubicación, entre los más importantes

El Sistema busca ser diseñado de tal forma que sea un genérico que pueda usarse en diferentes lugares, con los respectivos datos de una organización. Además se busca implementar el sistema a un bajo costo, pues existen en el mercado soluciones como las ofrecidas por Microsoft Surface[22] y PQ Labs[23] que son fabricantes de pantallas multitouch en el mercado y cuyos costos varían entre \$12500 - \$15000.

1.4 PROPUESTA DE SOLUCION

Se realizará una revisión del estado del arte, en la que se identifiquen Aplicaciones Tracker TUIO (Table-Top User Interfaces Objects) como las siguientes: reactIVision[3] , Touchlib [8], y CCV [9]. Además, se revisarán soluciones basadas en el rendimiento, multiplataforma, y precisión en la detección, bajo el protocolo TUIO [2] . Frameworks como PyMT (basado en Python), o Processing con su frameworktuoZone, interpretan de la misma forma los mensajes TUIO, que vienen desde el tracker.

Posterior a la etapa de revisión del estado de arte, se procederá a hacer un análisis y diseño de la solución y luego se procederá a seguir los siguientes pasos:

- Construcción del prototipo de pantalla multitouch.
- Implementación del Sistema que se usará en la Pantalla
 - Búsqueda de personas por medio de una nube de criterios de búsqueda, estos a su vez serían palabras representativas por las cuales se pueden crear asociaciones entre personas. Una vez encontrada se

puede proceder a visualizar en detalle información de la persona en cuestión.

- Búsqueda de personas por medio de palabras de interés, en la visualización de resultados se podrá observar a las personas y sus proyectos, además de la Unidad Académica a la que pertenecen.
- Pruebas de funcionalidad y usabilidad del sistema.

Conocido ya el problema que se espera resolver, decidimos que el lugar idóneo para ubicar el primer prototipo de pantalla sería en Rectorado General, debido a que este lugar posee un gran flujo de personas en el transcurso del día, y a su vez, el Sistema de información con datos de Rectorado serviría de ayuda a los visitantes. Otro lugar sería la entrada del PARCON ya que se necesitaría de un lugar de información, además de lo atractiva que puede llegar a ser una mesa colaborativa, con las funcionalidades de búsqueda descritas.

CAPITULO 2

2. REVISION DEL ESTADO DEL ARTE

Este capítulo resume los fundamentos teóricos y prácticos en los que se basó la solución propuesta para lograr el correcto funcionamiento del Sistema Multitouch, así mismo, se hace énfasis en las fortalezas de cada una de las alternativas de solución que se presentan.

2.1 PROTOCOLO TUIO

El protocolo TUIO es el formato del mensaje que define las coordenadas cartesianas de los *blobs* sobre la superficie de la pantalla. Un blob es un área rectangular convexa de píxeles activos, presentes en una imagen binaria [14]. Los blobs se dibujarán sobre la superficie de la pantalla, al momento del contacto de los dedos en ella. En otras palabras el protocolo TUIO, se puede definir como el protocolo de intercambio de información posicional de los dedos entre la aplicación servidor que envía los datos y la aplicación cliente que los recibe.

El protocolo nació en el 2005 y fue diseñado específicamente para interfaces tangibles al usuario. Inspirado por la idea de interconectar varias interfaces de mesas de colaboración tales como REACTABLE [2]. El protocolo TUIO ha sido implementado usando Open Sound Control. Existe una gran cantidad de implementaciones para diferentes lenguajes, como por ejemplo Java, C++, Pure Data, Flash, etc.

2.2 DESCRIPCION

El protocolo TUIO, permite estandarizar los mensajes que contienen la información de los objetos sobre la pantalla en el caso de la solución, los dedos. La forma de trabajar de este protocolo es definiendo dos clases principales de mensajes [2]: mensajes *set* y mensajes *alive*. Un mensaje *set* es el que tiene información sobre atributos del objeto sobre la superficie como la orientación, posición, etc. y un mensaje *alive* contiene información sobre un conjunto de objetos sobre la superficie, representados como una lista de ID's (Identificadores), cada objeto posee un id único.

Cuando sucede un cambio de estado en los mensajes set y alive se dan los siguientes procesos:

1. Los parámetros de los objetos colocados sobre la pantalla, son enviados después de un cambio de estado, usando un mensaje set.
2. Al remover un objeto de la superficie, un mensaje alive es enviado al servidor.
3. El cliente deduce adición y remoción de objetos a partir de los mensajes set y alive.
4. Los Mensajes fseqllevan el estado de la secuencia de objetos de un frame a otro con su respectivo conjunto de mensajes set y alive.

El envío de paquetes se realiza mediante el transporte UDP¹ (UserDatagramProtocol) que brinda una menor latencia en la comunicación, aunque existe la posibilidad que algunos paquetes se pierdan. La pérdida de paquetes se corrige incluyendo

¹Es un protocolo que ofrece una manera directa de enviar y recibir datagramas a través de una red IP, no ofrece un sistema de recuperación de paquetes pero es muy bueno cuando la velocidad es un factor muy importante en la transmisión.

información redundante. Una conexión TCP brinda un transporte más seguro pero a un costo más elevado de latencia.

2.3 ESPECIFICACIONES

El Formato de los mensajes está basado en Open Sound Control (OSC), `oscpack`² es una librería que recibe los siguientes tipos de mensajes con estos formatos[2]:

```
/tuio/[profileName] set sessionID [parameterList]
```

```
/tuio/[profileName] alive [list of active sessionIDs]
```

```
/tuio/[profileName] fseq int32
```

La Tabla 1, a continuación detalla los parámetros más importantes que refleja el objeto TUIO para una interfaz de superficie interactiva [2], parámetros directamente enviados desde la cámara (id, posición y ángulo) y otros calculados con el tiempo (velocidad y aceleración).

²<http://www.audiomulch.com/~rossb/code/oscpack/>

Tabla 1.Tipos semánticos de mensajes set en un objeto TUIO

Atributo	Característica
S	sessionID, temporary object ID, int32
I	classID, fiducial ID number, int32
x, y, z	position, float32, range 0...1
a, b, c	angle, float32, range 0..2PI
X, Y ,Z	movement vector (motion speed & direction), float32
A, B, C	rotation vector (rotation speed & direction), float32
M	motionacceleration, float32
R	rotationacceleration, float32
P	free parameter, type defined by OSC

2.4 OPEN SOUND CONTROL

Open Sound Control (OSC) [7] es un protocolo de comunicación que originalmente fue implementado para que los instrumentos musicales, ordenadores y otros dispositivos multimedia compartan datos de rendimiento de la música en tiempo real sobre una red.

2.4.1 DESCRIPCIÓN

OSC es un protocolo para la transmisión de datos musicales, expresados como un flujo continuo de paquetes (stream), que describen el estado instantáneo de la interfaz musical.

OSC se encuentra en la capa 6 de Presentación del Modelo OSI, el cual es similar a un lenguaje de intercambio de información como es XML [11].

Al igual que XML, OSC está diseñado para soportar una arquitectura cliente/servidor. Los datos son transmitidos en unidades de datos llamadas “paquetes”. El emisor de paquetes OSC es un *cliente* y el receptor es un *servidor*.

2.4.2 FORMATO

Los datos OSC se definen como mensajes y están formados por un patrón de direcciones (hacia dónde va dirigido el

mensaje), un tipo de dato, y argumentos (valores), un mensaje OSC se vería como en la Figura 1:

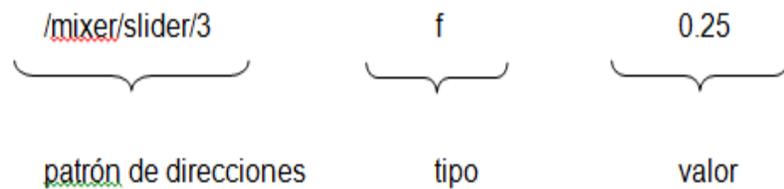


Figura 1. Formato mensaje OSC

Mientras que en XML se vería como en la Figura 2:

```
<?xml version="1.0"?>
  <message xmlns:xs="http://w3c.org/XMLSchema">
    <mixer>
      <slider />
      <slider />
      <slider type="xs:float">0.25</slider>
    </mixer>
  </message>
```

Figura 2. Formato de un mensaje XML

Como puede verse un mensaje definido con XML luce sobrecargado de etiquetas, mientras que OSC nos muestra una alternativa ideal en una aplicación de tiempo real.

2.4.3 IMPLEMENTACIONES

OSC tiene implementaciones tanto a nivel de hardware como de Software [13].

Dentro de la implementación a nivel de Software tenemos lenguajes como:

- C - Libro
- Matlab – Matlab OSC
- Java – Swing OSC Java Server
- Squeak – Squeak OSC
- .NET – OSC.net
- Python – Simple OSC
- Ruby – Ruby OSC implementation

Hay una gran variedad de lenguajes que soportan OSC lo que hace que este formato sea versátil y que se acomode a la preferencia de los desarrolladores.

2.5 TUIO TRACKER IMPLEMENTATIONS

El *tracker* es el programa que se ejecuta en el Servidor y que detecta la posición de los dedos sobre la superficie así como también su movimiento. Este programa hace uso de las tecnologías descritas en las secciones previas y se envía información de detección a un programa cliente que, por medio de un puerto, recibirá en el formato TUIO los atributos de cada uno de los blobs sobre la superficie.

Para el desarrollo de este trabajo se hará uso de trackers de código abierto, para en caso de ser necesario fácilmente hacer respectivos cambios al código. Existen una gran cantidad de ellos [15], cada uno tiene sus características particulares que los identifican.

2.5.1 REACTIVISION

Es un framework³ multiplataforma de visión por computadora especialmente diseñado para el rápido

³Conjunto de código el cuál provee una funcionalidad genérica la cual puede ser selectivamente sobrescrita o especializada por el programador proveyendo una funcionalidad específica.

desarrollo de interfaces de usuario en superficies tangibles, se basa en la detección de objetos con patrones adjuntos a ellos como se ve en la Figura 3, colocado sobre una superficie interactiva tangible, así como también el seguimiento de dedos sobre la pantalla.



Figura 3. Reconocimiento de Patrones

Reactivision es parte de ReactableSystems [16] que son superficies interactivas en las cuales se coloca objetos, según el patrón que el objeto tenga adjunto se emiten sonidos y se los relaciona con otros objetos sobre la pantalla. Actualmente el desarrollo de este software se da como un proyecto de código abierto.

Esta aplicación envía los mensajes TUIO vía UDP por el puerto 3333, a alguna aplicación cliente habilitada, también

puede enviar mensajes MIDI [17], estos parámetros pueden ser modificados en un archivo de configuración que el tracker provee. La Figura 4, describe la interfaz del Reactivision y sus archivos de configuración; en el área de la izquierda de esta figura se observa el archivo de configuración, en la parte superior derecha la imagen del tracker, y en la parte inferior derecha la consola de comandos; además existen implementaciones clientes en los siguientes lenguajes: C++, Java, C#, Processing, Pure Data, etc.

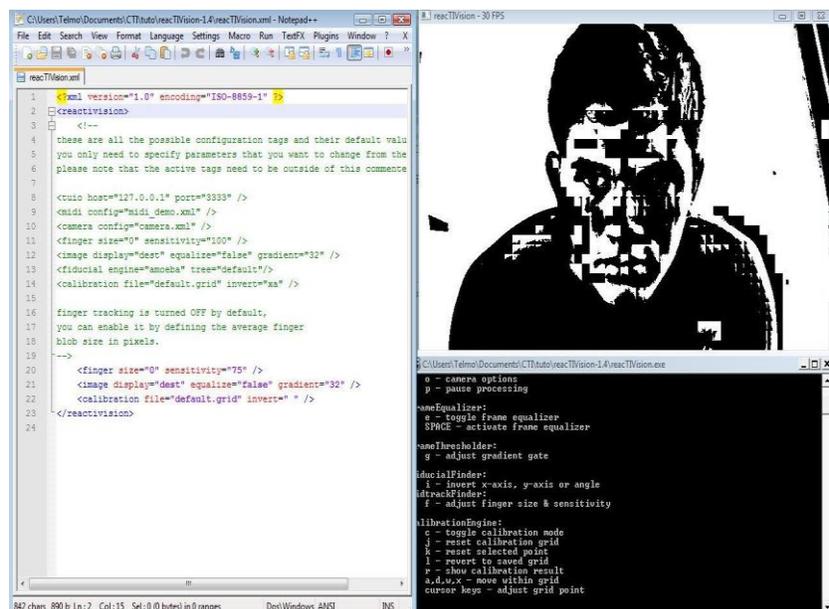


Figura 4. Archivos de Configuración, e interfaz Reactivision

2.5.2 Touchlib

Es una librería de código abierto, su principal funcionalidad es la del seguimiento de blobs [14] infrarrojos sobre una superficie, de esta forma detecta los dedos al moverse sobre la pantalla. Fue creada por el grupo NUI [8], la cual es una comunidad en línea, que se dedica al desarrollo de tecnología usando interfaces naturales al usuario, dentro de las cuales tenemos las pantallas multitouch. Actualmente solo trabaja en Windows lo que es una limitante de esta librería. En la Figura 5 se muestran las ventanas en cascada que Touchlib provee, como interfaz para el usuario, desde ellas podemos cambiar configuraciones del video como suavizado, brillo, contraste, etc.

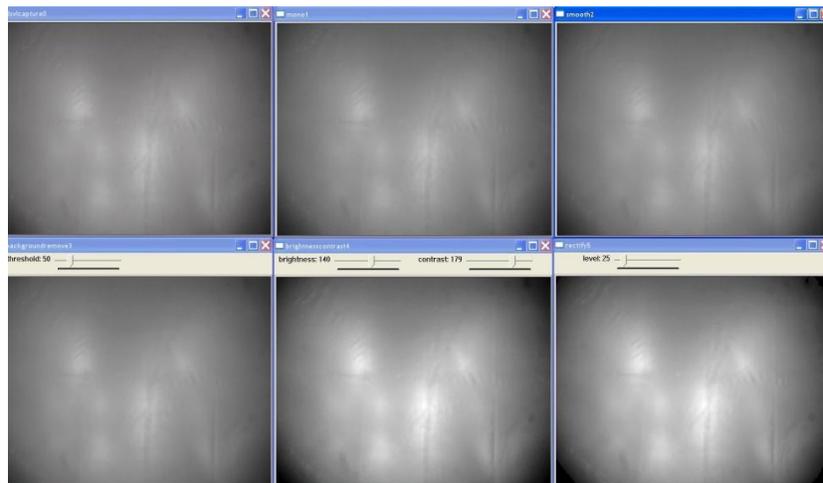


Figura 5. Interfaz de Touchlib

Touchlib envía mensajes TUIO vía UDP por el puerto 3333 a los clientes que estén recibiendo, lo que hace que este tracker sea compatible con aplicaciones clientes escritas en lenguajes como: Flash, VVVV, Processing, Pure Data, etc.

2.5.3 **CCV**

CCV (CommunityCoreVision) es una librería de visión por computadora, especializada principalmente en la detección de dedos sobre la superficie, al igual que touchlib fue desarrollada por el grupo NUI. La forma de detección puede

ser por medio de blobs infrarrojos o por blobs generados por la sombra que producen los dedos sobre la pantalla, bajo condiciones de luz ambiental uniforme.

La interfaz al usuario que provee CCV es mucho más amigable, haciendo que el aprendizaje para su buen uso sea sencillo. En relación a los dos casos anteriores, en el caso de Touchlib la cantidad de ventanas puede causar molestias y en el caso de reactIVision hay ciertas configuraciones que se las debe hacer por archivos de texto cuando pueden ser gráficas; una vez que se conoce cada uno de los parámetros de configuración en CCV, es más sencillo usar los otros dos.

CCV es multiplataforma y de código abierto, lo que posibilita el desarrollo de alguna característica adicional; CCV envía mensajes TUIO vía UDP por el puerto predeterminado 3333 a una aplicación cliente que este habilitada escuchando los mensajes. No solo se pueden enviar mensajes TUIO, también se pueden enviar mensajes XML o información

binaria. Por estándar la mayoría de implementaciones clientes escuchan mensajes TUIO.

La interfaz del programa tiene tres áreas bien definidas (ver Figura 6): en la región superior izquierda se pueden observar dos visores de videos, el uno el video fuente, y el otro el video modificado para el seguimiento de los dedos. En la región izquierda se pueden modificar ciertos parámetros de configuración, y por último la región inferior nos da la posibilidad de modificar parámetros del video.

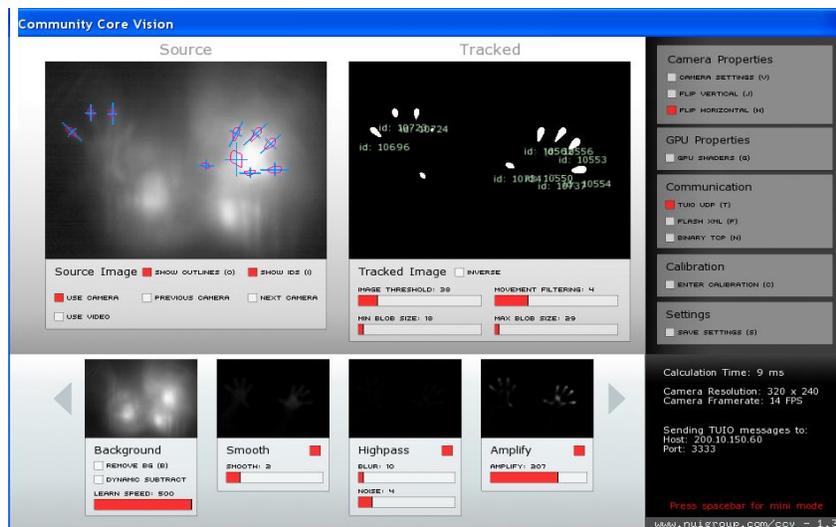


Figura 6. Interfaz de CCV

2.6 ANÁLISIS COMPARATIVO DE LIBRERÍAS DISPONIBLES

La tabla 2 ilustra las características mencionadas anteriormente con el objeto de apoyar el análisis comparativo de los diferentes Trackers presentados.

Tabla 2. Análisis de las características de los Trackers

CARACTERISTICA	REACTIVISION	TOUHLIB	CCV
Detección de blobs	X	X	X
Detección de patrones	X		
Multiplataforma	X		X
Código Abierto	X	X	X
Interfaz amigable			X

Como se puede observar el tracker idóneo para la implementación de una solución es CCV debido a la facilidad de uso y al buen rendimiento que posee; además, es un programa dedicado para el seguimiento de blobs, que es lo necesario para el correcto

funcionamiento de todo el Sistema. Sin embargo no se descarta en un futuro el uso de patrones en la pantalla multitouch para otro tipo de proyectos, de tal forma que se podría usar reactivision como tracker.

El uso de Touchlib es restrictivo a solo usar Windows lo que provoca una desventaja en términos de rendimiento con respecto a los que funcionan sobre Linux. Reactivision como función principal tiene la detección y seguimiento de patrones, lo que hace que CCV sea nuevamente ideal para el trabajo que se propone.

Existe un tracker sumamente nuevo que está siendo desarrollado bajo licencia QPL (Q PublicLicense)⁴, aún no está totalmente terminado pero se lo puede probar; este tracker tiene el nombre de **Movid**[24]. Posee características en común con algunos de los antes mencionados; como por ejemplo, que es multiplataforma, y características propias que resaltan tales como: el uso de varios cores para el procesamiento, y la administración web. En la Figura 7 se puede observar la Interfaz de Administración Web; las cajas naranjas al lado derecho del video son módulos del sistema. A

⁴Es una licencia en la que no se puede distribuir legalmente el producto, es incompatible con GPL.

diferencia de los trackers estudiados que son standalone (se lo instala en un pc y se lo administra desde allí), éste se lo puede administrar a través de la red por medio de una interfaz HTML5. La configuración del tracker es modular, adaptándose a las necesidades del programador, es decir en caso de necesitar un tracker que siga objetos en la pantalla, lo único que se hace es agregar los módulos camera, tracking, tuio, etc. En la Figura 8 cada línea del archivo de texto define un módulo que se representa por una caja naranja en la interfaz que aparece en la Figura 7. Y en caso de necesitar detección de patrones se agrega el modulofiducial. Aun no existe un release de este software pero sin duda alguna hemos considerado que por las características que tiene era de importancia mencionarlo.

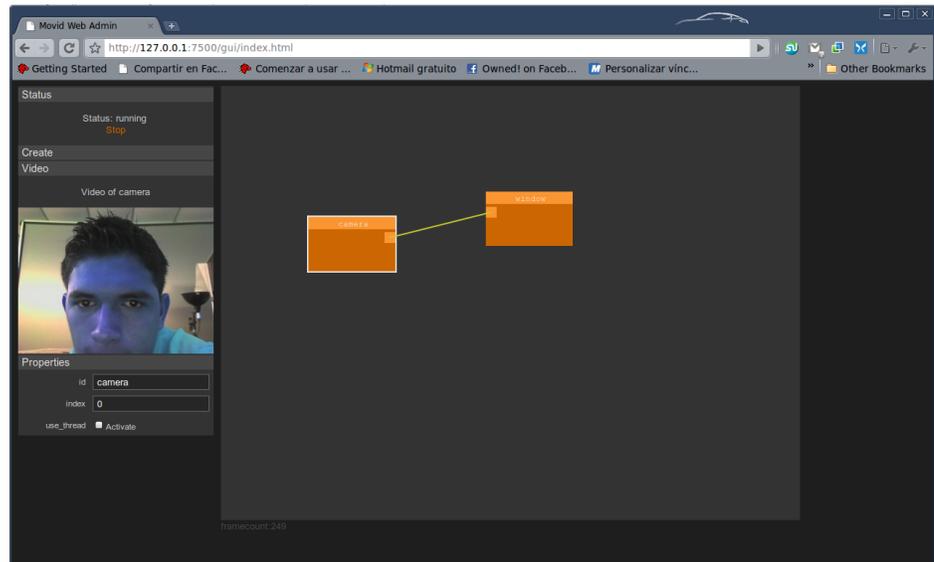


Figura 7. Administración Web Movid

```

andres@andres-lap: ~/Movid/presets
File Edit View Terminal Tabs Help
andres@andres-lap: ~/wor... andres@andres-lap: ~/Mov...
# Fiducial test on camera
#
# create defaults objects
pipeline create Camera camera
pipeline create GrayScale gray
pipeline create Threshold threshold
pipeline create Smooth smooth
pipeline create FiducialTracker tracker

# do connections
pipeline connect camera 0 smooth 0
pipeline connect smooth 0 gray 0
pipeline connect gray 0 threshold 0
pipeline connect threshold 0 tracker 0

# debug
#pipeline create Dump dump
#pipeline connect tracker 1 dump 0

# combine image source with image from tracker
# and show in a window
pipeline create ImageDisplay display
pipeline create Combine combine
pipeline connect camera 0 combine 0
pipeline connect tracker 0 combine 1
pipeline connect combine 0 display 0

pipeline create Tuio tuio
pipeline connect tracker 1 tuio 0

pipeline set camera use_thread 1
pipeline set gray use_thread 1
pipeline set threshold use_thread 1
pipeline set smooth use_thread 1
pipeline set tracker use_thread 1
pipeline set combine use_thread 1
pipeline set tuio use_thread 1

1,1 All

```

Figura 8. Presets personalizables según necesidad de programador

CAPITULO 3

3. ANÁLISIS Y DISEÑO DE LA APLICACIÓN

Este capítulo abarca una sección en la cual se realiza un análisis y diseño de la solución propuesta, así como, los procedimientos, cálculos, y métodos empleados para lograr el funcionamiento de la pantalla multitouch, además de la tecnología y herramientas usadas a nivel de software para la implementación de la aplicación para el Hardware. Se realizará además una descripción de cada uno de los módulos del Sistema de Búsqueda, y se terminará el capítulo con el Plan de Pruebas.

Entre los requerimientos que se cumplirán dentro del alcance del Sistema se tiene:

- Requerimientos Funcionales
- Requerimientos No Funcionales

Requerimientos Funcionales

Los requerimientos funcionales definen cuales son las funciones que el usuario realizará a medida que use la aplicación.

1. Interacción a través de Gestos: Forma de interactuar por medio de uno o más toques, además de su dirección y velocidad sobre la pantalla.

Una de las premisas principales en una aplicación multitouch es el buen uso de gestos sobre la pantalla como forma de interacción. Para ello en los siguientes gráficos presentaremos en detalle los comúnmente usados.

Al presionar con nuestro dedo sobre la pantalla y levantarlo es como que generamos un evento parecido al dar clic izquierdo con el mouse.

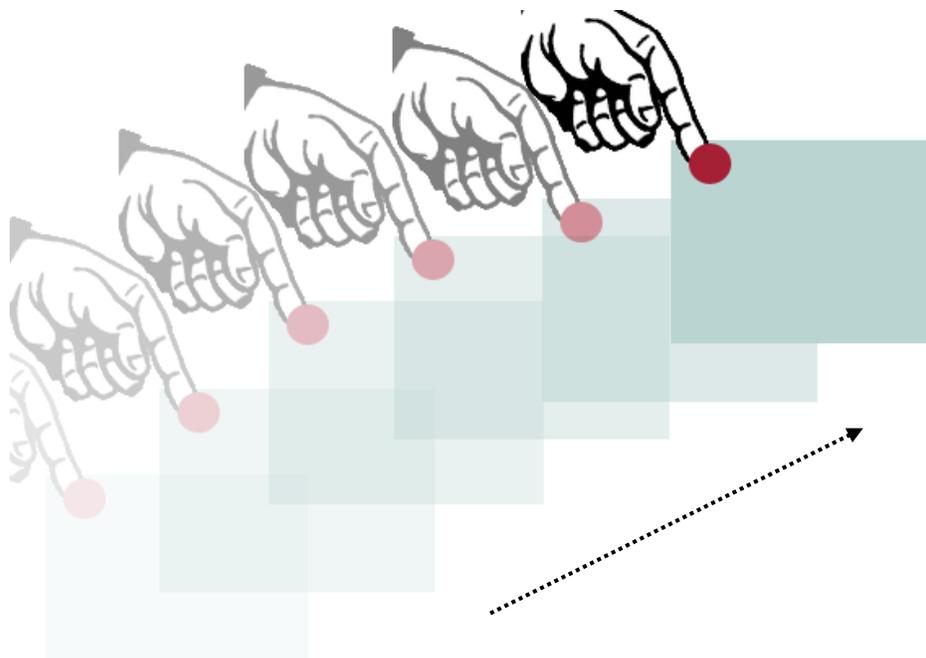


Figura 9. Gesto de Arrastrar y Soltar

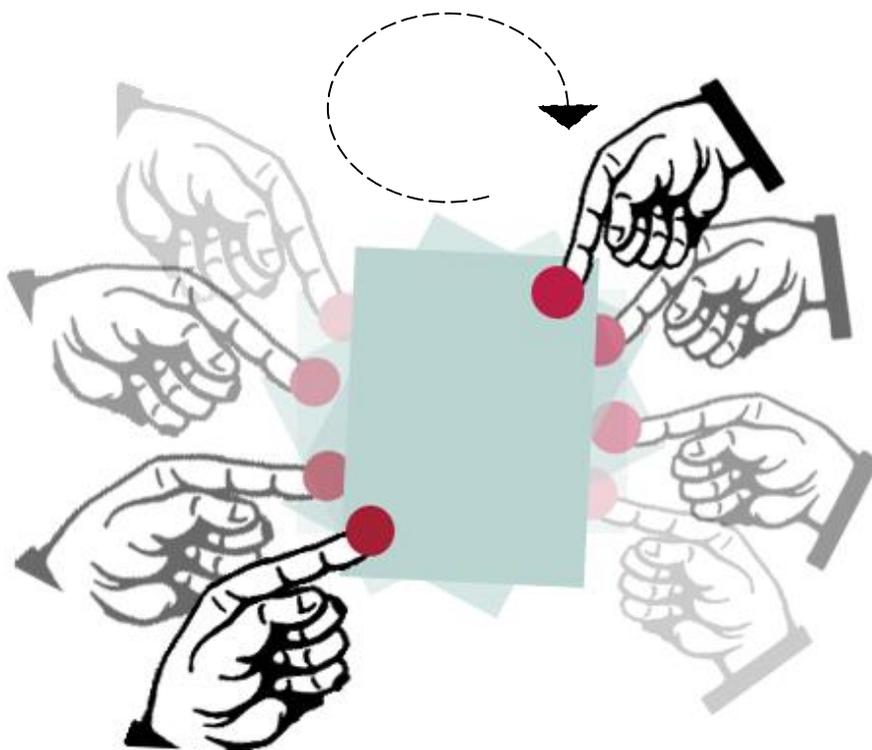


Figura 10. Gesto de Rotar

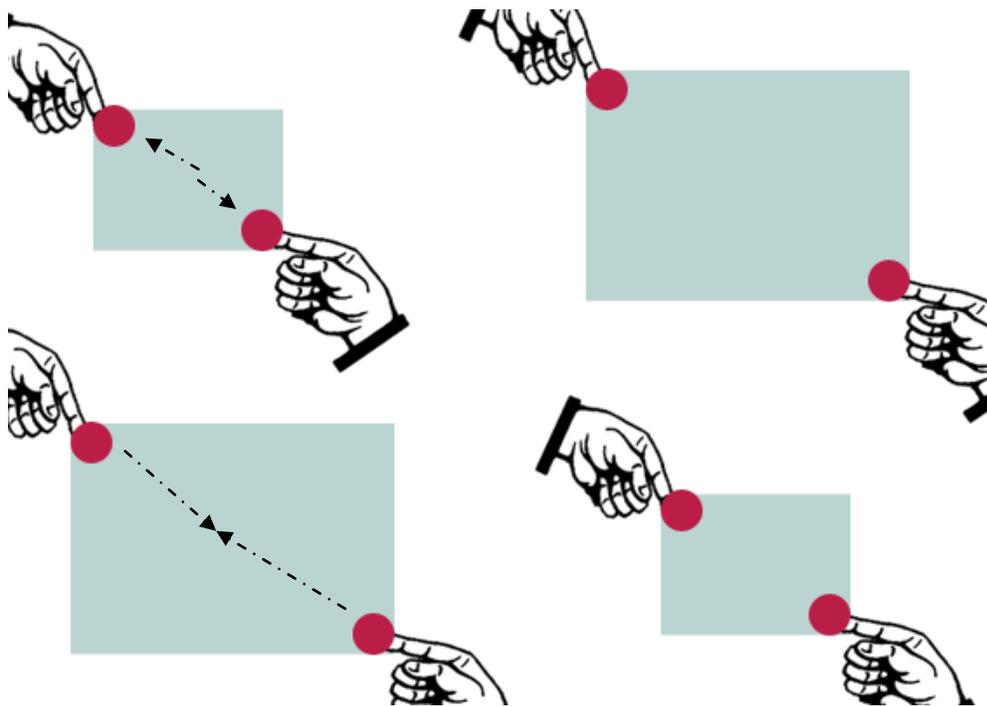


Figura 11. Gesto de Escalar

En la Figura 9 notamos que podemos mover un objeto a lo largo de una trayectoria definida por nuestros dedos. En la Figura 10 y Figura 11 notamos ya un gesto en el que necesitamos al menos de dos toques; en la Figura 10, el ángulo en el que rotemos nuestros dedos, será igual al que el objeto rote sobre su eje, y en la Figura 11, al acercar nuestros dedos el objeto se hace más pequeño, y al alejarlos se agrandan. Además de estos gestos predefinidos, se pueden personalizar gestos según el gusto del programador, por ejemplo el hecho de formar un

cuadrado, o un círculo puede disparar un evento y hacer algún cambio en la aplicación.

- 2. Interacción a través de teclado en Pantalla:** Se proporcionará un teclado en pantalla como el que observamos en la Figura 12, para objeto de ingresar información



Figura 12. Teclado QWERTY Táctil

- 3. Seleccionar Palabras Claves:** Le permite al usuario seleccionar la palabra o palabras que permitirán iniciar la búsqueda; estas palabras estarán disponibles al usuario, tal como lo ilustra la Figura 13.



Figura 13. Palabras Claves

- 4. Ver Detalles de Persona:** Se mostrará una ventana flotante, con información de la persona como se ve en la Figura 14.

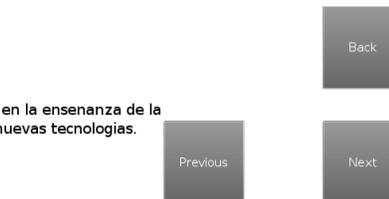
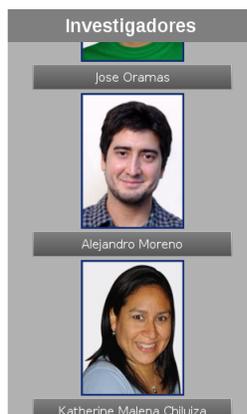


Figura 14. Detalles de Personas

- 5. Presentación de Proyectos de Investigación y personas asociadas:** En la Figura 15 se ilustra lo que se espera presentar en relación a este requisito, esto es, detalles de proyectos realizados por una persona seleccionada y las personas que colaboraron en él.

Hand Gesture Recognition Interface

Probar la eficiencia de las metodologías empleadas actualmente en la enseñanza de la audición deteriorada de los niños mediante la incorporación de nuevas tecnologías.



Jose Oramas M., Alejandro Moreno C., Katherine Chiluita G.

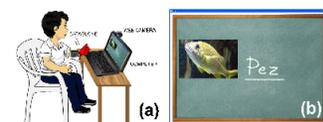


Figure 1: Interacting with the system

4.1.2 Gesture Recognition

Gestures are the trajectories or allocation made while moving the hands. For the recognition of the gestures, the location of the hand is required at every moment of the interaction in order to identify its trajectory. In the proposed system, hand tracking is done by color segmentation algorithm. One step prior to the segmentation is to cover the cyborgs with a red glove then, using a web camera, images of the user are continuously captured. These images are changed to the Hue-Saturation-Value (HSV) color space and the glove is detected using color segmentation algorithms filtered with erosion/dilatation operations. The main reason the HSV color space is selected was that it emphasizes the human perception in hue and that it has an easily invertible transform from RGB [11, 12] which makes it fast enough for real time processing. Once the coordinates of the hand, defined by the centroid of the red blob, are obtained. These coordinates are sent to a Grid Algorithm based on the algorithms proposed in [5, 9], for stroke recognition.

4.1.3 Sign Recognition

Once postures and gestures are detected, they are compared against a set of rules. This set of rules describes all the signs recognizable by the system. As a measure to deal with variation or noise introduced by users

Figura 15. Proyectos y Relación de Personas

Requerimientos No Funcionales

Los requerimientos no funcionales influyen en la operatividad del Sistema, hemos considerado los siguientes:

1. **Precisión:** La detección y seguimiento de los dedos, debe calzar con lo que se muestra en pantalla como coordenada **x** y **y**.
2. **Rendimiento:** La respuesta a las acciones deben ser rápidas de tal forma que el usuario sienta un ambiente natural de uso.
3. **Apariencia:** La interfaz debe ser de uso intuitivo y sencillo, que no requiera experticia del usuario. Debido a la proyección que se

tiene que realizar, los colores a usar son fuertes y oscuros, negro, azul etc.

Arquitectura

Primero se presenta la arquitectura de la Pantalla Multitouch, y luego la del sistema completo.

Por medio de software basado en algoritmos de inteligencia artificial y visión por computador que se capturarán a través de una aplicación de tracker de TUIO, se detectarán los movimientos que se den sobre la superficie. Estos movimientos se denominan Eventos Multitouch, los que permiten realizar acciones indicadas por el usuario. La Figura 16 esquematiza lo explicado y presenta de forma sencilla la arquitectura de la Pantalla Multitouch.

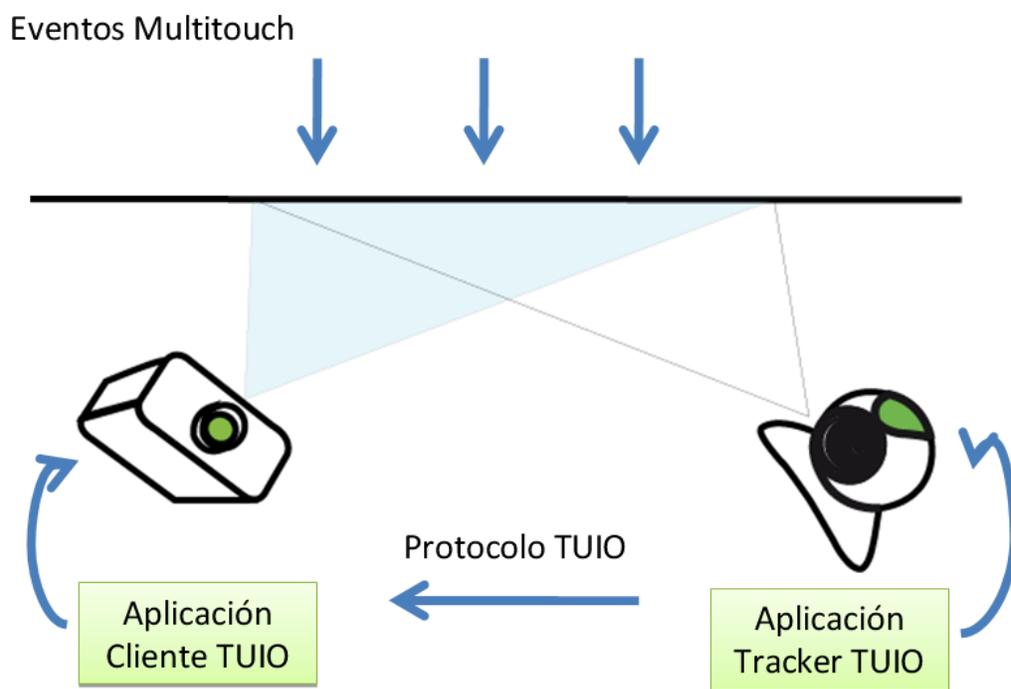


Figura 16. Esquema de Funcionamiento de la Pantalla Multitouch

El Sistema propuesto está conformado de tres capas, la Figura 17 describe las capas del sistema.

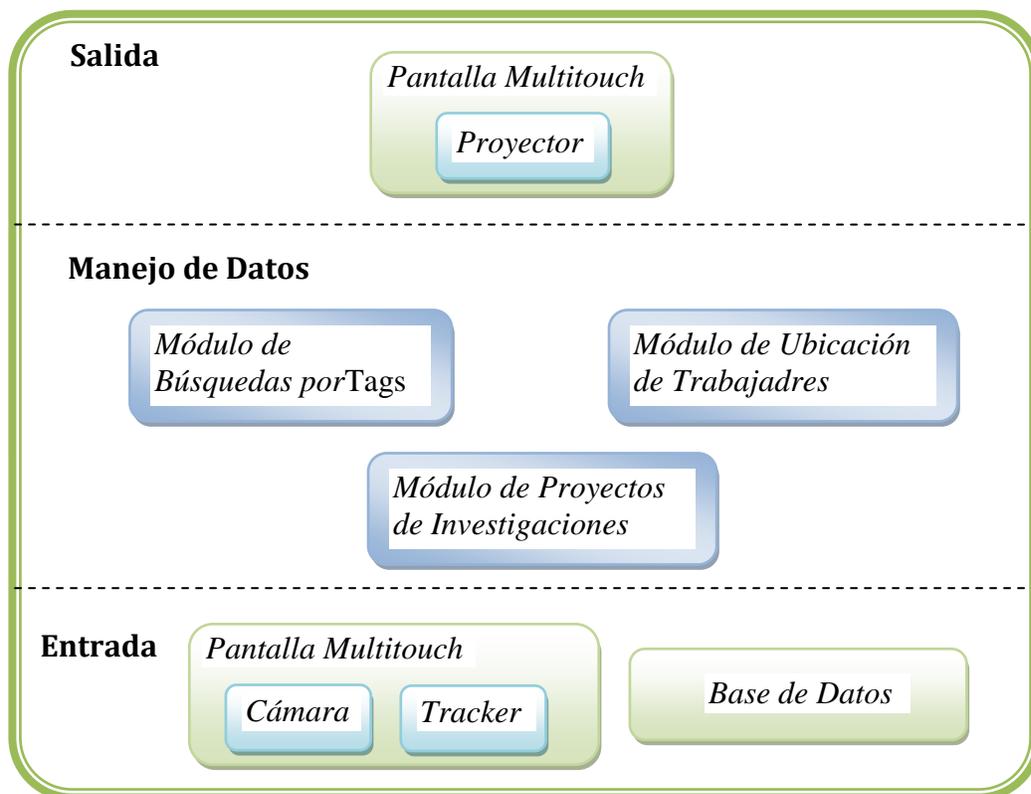


Figura 17. Arquitectura del Sistema

Como se puede observar en la Figura 17, en la **capa de entrada** tenemos la Base de Datos del Sistema la cual lo alimenta con información de los integrantes de un Centro de Investigación o de toda la ESPOL, como dispositivo de entrada y salida se tiene a la Pantalla Multitouch, la que por medio del Tracker envía la información necesaria para ser procesada por el Manejador de Datos. La Base de Datos

contendrá información referente a: personas, proyectos, fotos, ubicación, y palabras claves que relacionen personas, proyectos, fotos entre sí.

El manejador de datos consiste de una aplicación cliente que recibe los datos enviados por la capa de entrada, los procesa y carga toda la información en la aplicación. Una vez que el manejador de datos ha concluido su trabajo, la información se presenta en la pantalla multitouch.

En la capa de manejo de Datos se hará uso de las librerías que permitan hacer una aplicación que reciba mensajes TUIO; además, de la diagramación de la interfaz gráfica a implementar. La decisión de qué librería usar se realizará en el capítulo 4. Se ha dividido la capa de manejo de datos en tres módulos en cada uno de ellos se manejan los eventos que se producen al realizar acciones determinadas. El módulo de Búsqueda de Tags, el módulo de Ubicación de Trabajadores, el módulo de Proyectos realizados.

El módulo de Búsqueda de Tags despliega una lista de palabras claves que nos servirán para encontrar personas y proyectos por medio de ellas.

El módulo de Ubicación muestra un mapa con un camino desde rectorado al lugar de trabajo de la persona.

El módulo de Proyectos realizados muestra una lista de los proyectos, sobre los cuales se puede saber detalles como una descripción y personas involucradas.

La capa de Salida es la presentación del Sistema al usuario, al igual que en la Entrada es la Pantalla Multitouch pero en este caso tenemos al Proyector como medio de interacción con la pantalla.

Diseño de la Aplicación que maneja los datos



Figura 18. Selección de palabras claves

En la Figura 18, se diferencia áreas de criterios de búsqueda, otra de presentación de resultados, etc.

En la Figura 19 se definen las regiones de la interfaz, donde el usuario interactuará.

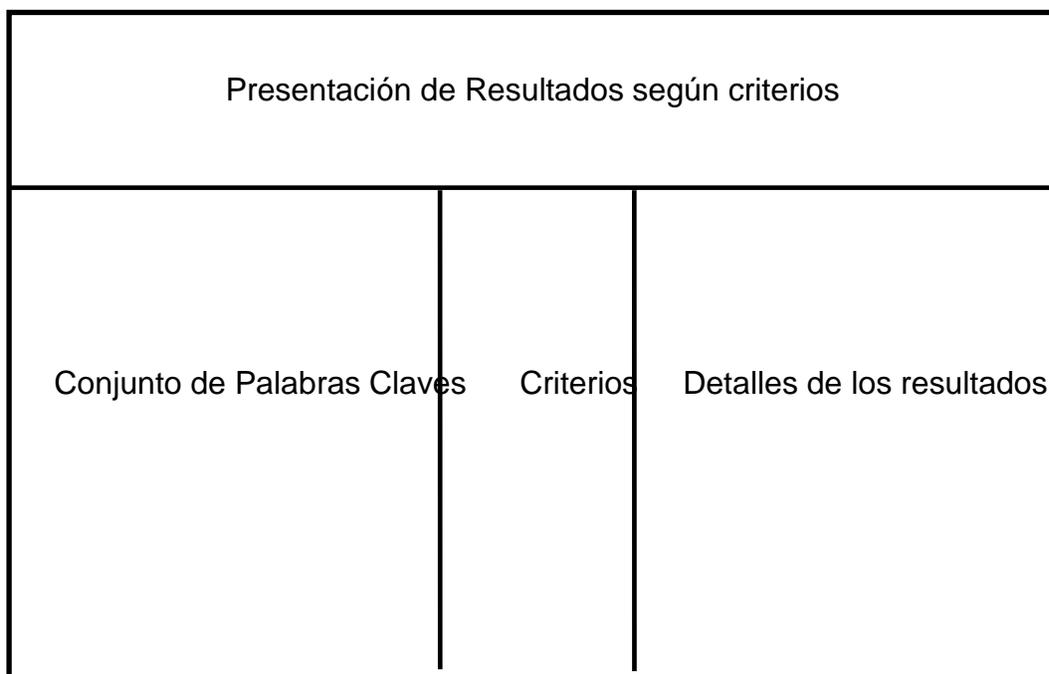


Figura 19. Diseño de la Interfaz.

Se enviarán las distintas palabras claves (temas, nombres de personas o proyectos, etc.) al área de criterios de búsqueda, según la(s) palabra(s) seleccionadas los resultados se filtraran en el área de

presentación. El sistema retornará ubicación de una persona dada (nombre del edificio, teléfono o extensión, oficina) y una ruta(s) sugerida(s) a partir del rectorado (caso específico de este proyecto), esta información se muestra en el área de detalles de resultados.

En la siguiente sección se presenta el proceso de análisis y diseño de la pantalla multitouch, dispositivo al que llamaremos *mesa* de ahora en adelante; luego seguiremos con la descripción de cada uno de los módulos de la capa de Manejo de datos.

3.1 ANÁLISIS Y DISEÑO DE LA PANTALLA

Este proyecto ha sido realizado utilizando 3 prototipos de prueba, con el objeto de seleccionar la tecnología adecuada para realizar la implementación de la solución.

Primer Prototipo

El primer prototipo, se centró en la creación de la posible pantalla de interacción. En este prototipo se usó una caja de cartón de 15cm x 15cm, una de sus caras se la cubrió con papel plano y se colocó una lámina delgada de acrílico encima como se muestra en la Figura 20, se colocó una webcam en el fondo de la caja, se la

conectó a la computadora, y por medio del tracker se hizo el respectivo seguimiento de los dedos.



Figura 20. Primer Prototipo de la Pantalla Multitouch

Para el primer prototipo los materiales usados fueron los que observamos en la Tabla 3.

Tabla 3.Lista de Materiales Primer Prototipo

Cantidad	Material
1	Caja de Cartón
1	Lámina de Acrílico 2 mm de ancho
1	Cámara webGenius 320x240
1	Material Difusor: papel plano

Este prototipo sirvió para definir una arquitectura de la mesa a pequeña escala; además del uso de CCV como solución para el seguimiento de los dedos. Se trabajó por medio de sombras para la detección de los blobs, el seguimiento de los dedos dependía de la cantidad de luz ambiental.

Segundo Prototipo

Para el siguiente prototipo ya se estableció construir el mueble de la mesa, para lo cual se hizo un análisis sobre las dimensiones a usar. Inicialmente se consideraron dos principales factores para tomar la decisión acerca de las medidas, el primero la cantidad de personas que puedan simultáneamente de una forma cómoda, usar la mesa; el número escogido fue 4, y el segundo las limitaciones con respecto a hardware con que contamos, es así que llegamos a la Tabla 4.

Tabla 4. Lista de Dispositivos del prototipo 2 y características de resolución y medida de proyección

Dispositivo	Resolución	Medida de Proyección
webcam	320 x 240	(61 +/- 2) cm x (45+/- 1) cm
webcam	640 x 480	(105 +/- 3) cm x (75+/- 2) cm
Proyector	1280 x 800	(70+/-1) cm x (51+/-1) cm

Las medidas de proyección de la Tabla 4 se las tomaron a 90 cm de distancia que es la medida de alto que tendrá la mesa, si bien es cierto el área de proyección del infocus es mucho menor en relación al de la cámara de 640 x 480, este problema se lo buscará resolver con el uso de espejos, ya que por principios de óptica a mayor recorrido de un haz de luz, mayor es su área de proyección.

Muller [25] y la comunidad NUI Group [4], sugieren que las medidas adecuadas para el área de trabajo de la mesa son **95 cm x 72 cm**, y esta medida fue la adoptada para la construcción del Tablero de colaboración, como observamos en la Figura 21 y la Figura 22.



Figura 21. Vista superior de la mesa

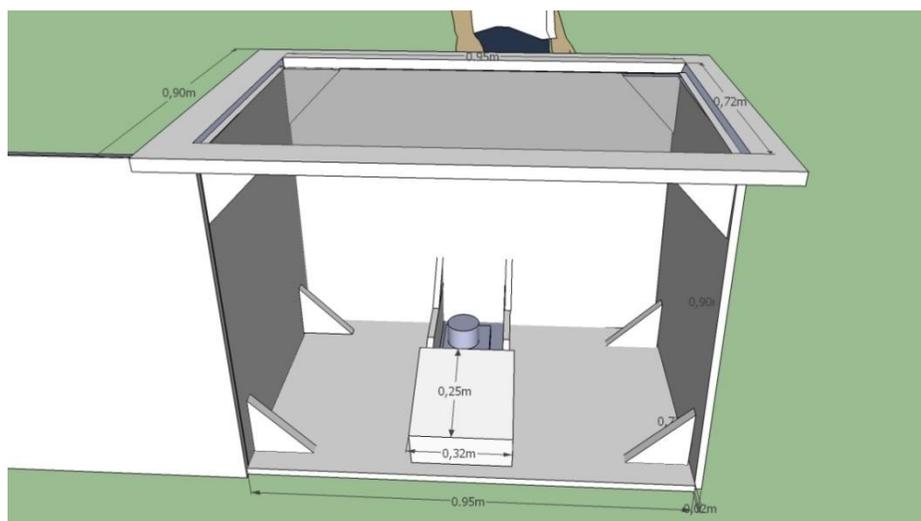


Figura 22. Vista lateral de la mesa

Una vez resuelto el problema de las medidas, la siguiente decisión consistió en definir el tipo de pantalla multitouch a construir. Las

diferentes implementaciones existentes [4] [25], consideran que para el desarrollo del Sistema de Búsqueda la mejor opción es **FTIR** (FrustratedInternalReflection). FTIR consiste en tener leds infrarrojos alrededor de la superficie de proyección, de tal forma que, al presionar sobre la superficie la luz se refleje contra la cámara, la cual es una solución ideal para el seguimiento de blobs infrarrojos. La solución que se decidió implementar fue **Rear DI** (DiffusedIllumination), que al contrario de lo que sucede con FTIR los leds se los coloca debajo de la superficie con el objetivo de lograr una iluminación uniforme. Es decir, que al presionar la pantalla, la luz que atraviesa el acrílico rebote generando un blob. Esta solución si bien requiere de mayor procesamiento que FTIR, también posee una ventaja que es la detección de patrones; de esta forma, en un futuro se podría implementar algún sistema usando esta tecnología.

En la Figura 23 encontramos un esquema de la solución por Iluminación Difusa.

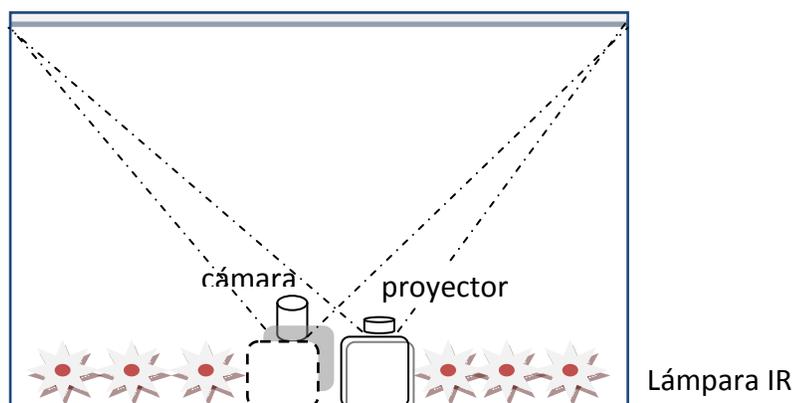


Figura 23. Esquema de una mesa Rear DI

El siguiente componente de la mesa a ser analizado es el material con el que se construirá la superficie de interacción. El acrílico, a diferencia del vidrio es un material que resiste más los impactos y el ancho es de 9 mm para evitar que al apoyarse sobre la superficie esta se pueda deformar o romper.

Debajo de la superficie de interacción se utiliza un material difusor de luz que a su vez, sirva para proyectar sobre él, en este prototipo se usó papel plano como difusor.

Por último se analiza la cámara que se usó, la cual fue una Genius con una interfaz USB 1.1 que maneja velocidades de transmisión de información que van de los 1.5 Mbits/s a los 12 Mbits/s.

En la Tabla 5 presentamos la lista de materiales usados para el Segundo Prototipo.

Tabla 5. Lista de Materiales del Segundo Prototipo

Cantidad	Material
1	Mueble en forma de mesa cerrada
1	Lámina de Acrílico 9 mm de ancho
1	Cámara web Genius 320x240
1	Material Difusor: papel plano
1	Proyector

Las pruebas realizadas con este prototipo involucraron el uso de luz ambiental como forma de iluminación, debido a la dificultad de conseguir lámparas de luz infrarroja en el medio local.

El tiempo de cálculo para el tracking de los dedos estuvo entre los 9 y 12 ms, los frames por segundo estuvieron entre los 12 a 14, a una resolución de 320 x 240.

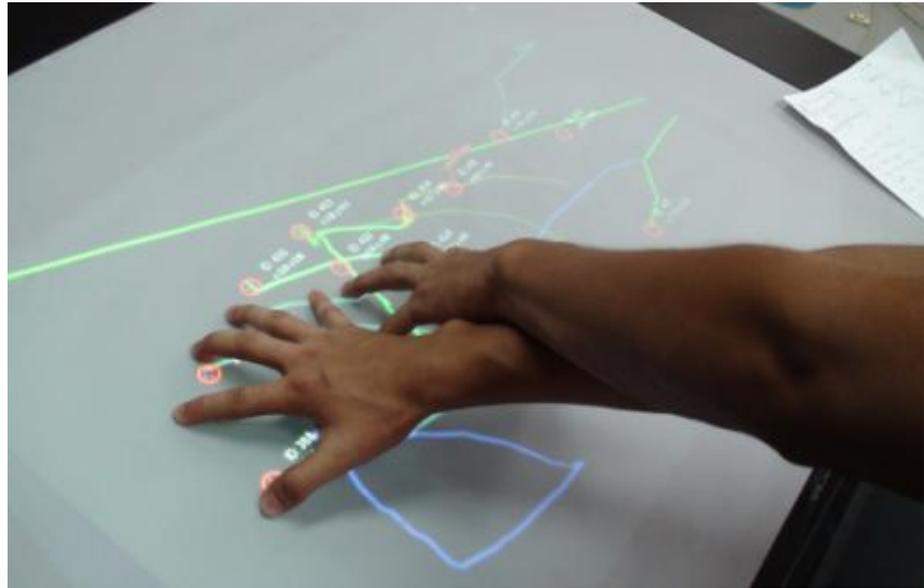


Figura 24. Prueba de precisión Segundo Prototipo

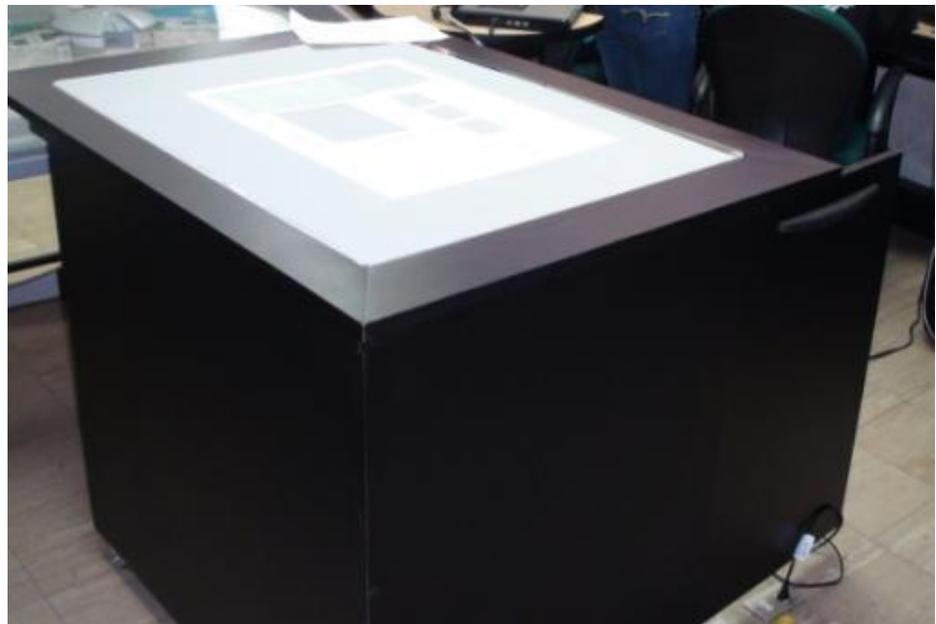


Figura 25. Segundo Prototipo

Dentro de los problemas que se encontraron estuvieron la falta de precisión al momento de usar la Pantalla como observamos en la Figura 24, debido a la mala calibración del Tracker, si bien los resultados no fueron tan precisos y había latencia de 0.5 segundos en la comunicación cliente – servidor; es decir, el seguimiento de los dedos no era en tiempo real. La resolución de 320 x 240 generó una región 50% menor que la esperada y por último el uso de papel plano como difusor provocaba una mala detección de los dedos, debido a que este material absorbía mucha luz. Bajo todos estos indicadores este fue un buen inicio para la investigación y levantamiento de datos. La Figura 25 presenta el Sistema completo en el Segundo Prototipo.

Tercer Prototipo

En el tercer prototipo tenemos cambios significativos, entre los que se destacan el uso de luces infrarrojas como medio de iluminación, el cambio del material difusor de luz, la cámara usada, etc. La lista final de materiales usados se la puede observar en la Tabla 6, así como también los costos de de éstos.

Tabla 6. Lista de Materiales y Costo del último Prototipo

Cantidad	Material	Costo
1	Mueble en forma de mesa cerrada	\$350
1	Plancha de Acrílico de 96cm x 73cm	\$145.54
1	Proyector	\$600
12	Arreglo de 20 leds infrarrojos	\$108
2	Fuentes	\$5
1	Adaptador de 12 voltios	\$5
1	Cámara Ps3Eye modificada	\$60
1	Espejo 60cm x 60cm	\$14
1	Material Difusor: cobertor de mesa de plástico color blanco	\$2
Total		\$1289.54

Luego, de un análisis sobre las necesidades de que la luz atraviese la superficie, se decidió usar un material plástico que se usa a manera de mantel sobre superficies. Una característica importante del material es que tiene diminutos poros, por donde la luz se filtra y además es una buena superficie sobre la cual proyectar. En la Figura 26 observamos el material difusor usado.



Figura 26. Material Cobertor de Mesas.

La inclusión de una cámara con mejores características fue necesaria. Por ello la ps3eye, que si bien no es óptima para seguimiento, ni posee una buena resolución, es una cámara con características aceptables y que se puede adaptar a este Sistema. La ps3eye (ver Figura 27), funcionó relativamente dentro de lo esperado a 640 x 480 y ocupa el área total de la pantalla, a 30 fps logrando no perder el seguimiento de los dedos con movimientos rápidos, y con tiempos de procesamiento en el tracker de 14 a 20 ms.



Figura 27. PS3EYE.

El funcionamiento de la cámara fue el ideal, pero aún no fue suficiente, ya que al hacer detección de blobs con luz ambiental y, el proyector funcionando al mismo tiempo el seguimiento de los dedos se perdía (debido a que la luz ambiental y la del proyector trabajan en la misma longitud de onda de la luz visible). La solución a este problema fue trabajar la cámara y el proyector a diferentes longitudes de onda, la cámara necesitó ser modificada, de forma que sea una cámara de luz infrarroja. Este cambio requirió usar lámparas infrarrojas que iluminan la superficie. La Figura 28, muestra una de las lámparas usadas en la implementación del prototipo 3.



Figura 28. Lámparas de Luz Infrarroja.

En el país no fue posible encontrar lámparas infrarrojas, por ello, se implementó un arreglo de leds infrarrojos, El número total de lámparas que se construyeron fue 12 con lo cual se logró la iluminación requerida. Estas lámparas son realmente placas de leds infrarrojos y su diseño se lo observa en la Figura 29.

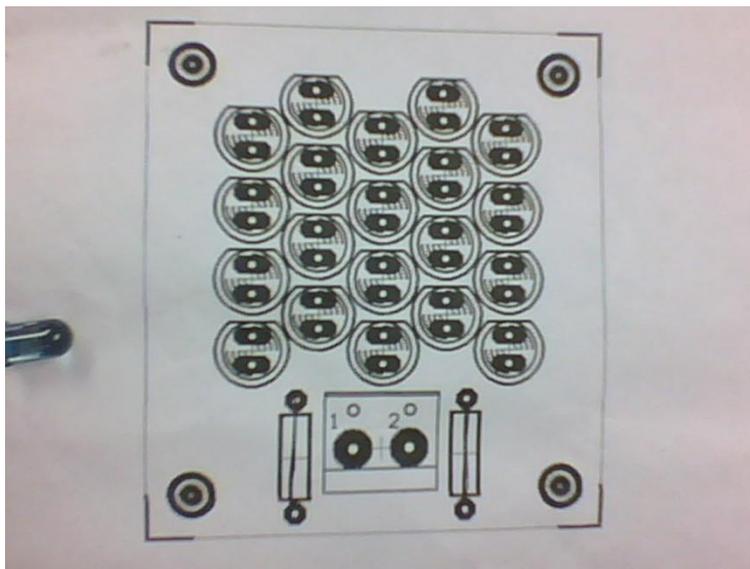


Figura 29. Esquema de las lámparas de Luz Infrarroja.

El proyector usado tiene una resolución de 1280 x 780, para alcanzar la máxima área de proyección se hizo uso de un espejo.

Una vez que se realizó el análisis y diseño de la arquitectura y elementos que presentará la pantalla multitouch, se procede en el siguiente capítulo a la implementación del dispositivo, procesos como calibración y colocación de los elementos se mostrará en el siguiente capítulo.

3.2 MÓDULO DE BÚSQUEDA POR TAGS

El Módulo de búsqueda es aquel que utilizará un conjunto de palabras (palabras claves) a las cuales llamaremos de ahora en adelante tags, a las que se les asociará un conjunto de elementos para realizar la búsqueda a partir de ellas. Esta búsqueda consistirá de arrastrar a un área de criterios los tags para luego visualizar los resultados asociados a ellos.

En la Figura 30 se muestra un esquema gráfico del módulo, del cual tenemos como entrada los datos de la Tabla Keyword, el cuerpo del Módulo se compone de dos clases que manejan la parte de presentación de los datos, como salida tenemos una lista con los Tags que han sido seleccionados.

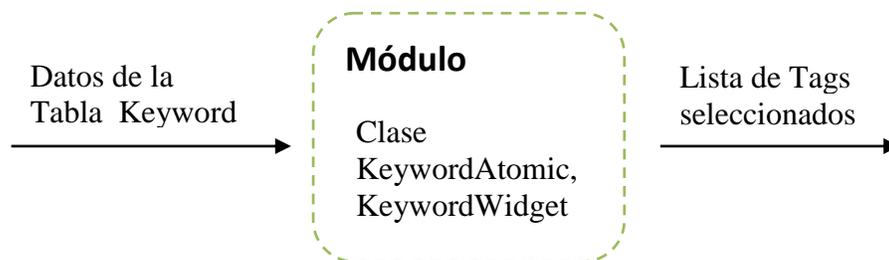


Figura 30. Esquema E/S del Módulo de Búsqueda por Tags.

Los tags según su importancia tienen un tamaño de fuente, un color para diferenciarlos y se muestran dispersos sobre el área de tags.

3.3 UBICACIÓN DE TRABAJADORES

Como se indicó en la sección previa uno de las funcionalidades del módulo es encontrar personas. Una vez que el sistema identifica a quién se busca, se procede a presentarlo y se presenta la ubicación de dicha persona.

Como vemos en la Figura 31 tenemos como datos de entrada el identificador de la persona, y los datos de la Tabla Ubicación que los obtenemos de la relación en las Tablas persona y ubicación. Se implementa una clase que maneja estos datos de entrada y da como resultado de salida una imagen con datos procesados.

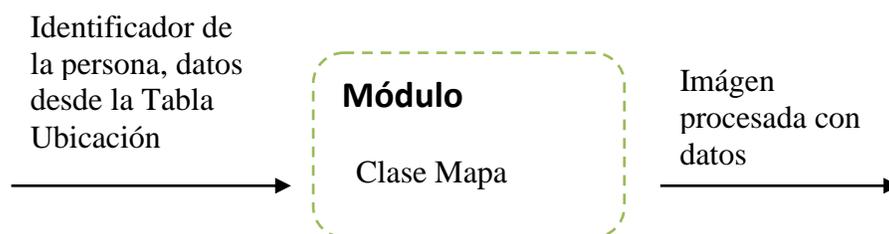


Figura 31. Esquema E/S del Módulo de Ubicación de Trabajadores.

Se podrá escalar, rotar el mapa (Figura 32) a disposición del usuario, con los gestos universales empleados en dispositivos táctiles como por ejemplo el Iphone.



Figura 32. Mapa de la ESPOL

3.4 PROYECTOS DE INVESTIGACIÓN REALIZADOS

Otro de los detalles del resultado de la búsqueda es la presentación de los proyectos realizados (nombre y descripción), además de las personas involucradas con estos. Para este módulo

tenemos como entrada el Identificador de la Persona, y la Tabla Project de donde obtenemos los datos necesarios para obtener la Lista de Proyectos (Figura 33)

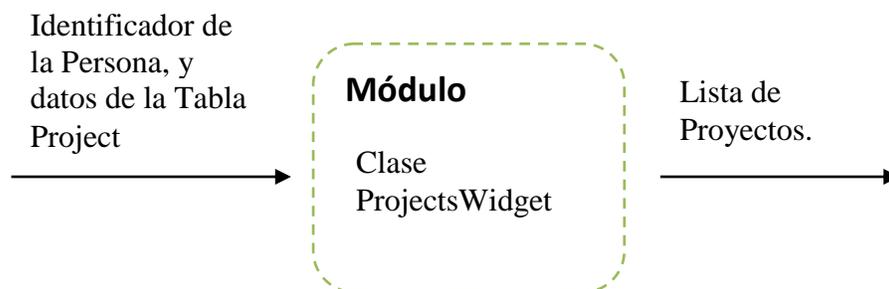


Figura 33. Esquema E/S del Módulo de Proyectos de Investigación.

3.5 RELACIONES DE DEPENDENCIA ENTRE PERSONAS Y PROYECTOS

Cada proyecto a su vez posee una lista de integrantes de las personas que trabajaron en dicho proyecto. Esta lista es mostrada en la descripción del proyecto. Se busca relacionar a las personas involucradas según el proyecto que hayan realizado juntos. La información a mostrarse es el nombre del investigador y su foto.

Diseño de la Base de Datos

Otro componente importante de la capa de entrada es la Base de Datos que contiene la información que se requiere mostrar al momento de las

búsquedas. Para esto se ha elaborado la siguiente estructura para la Base de Datos que se la puede observar en la Figura 34.

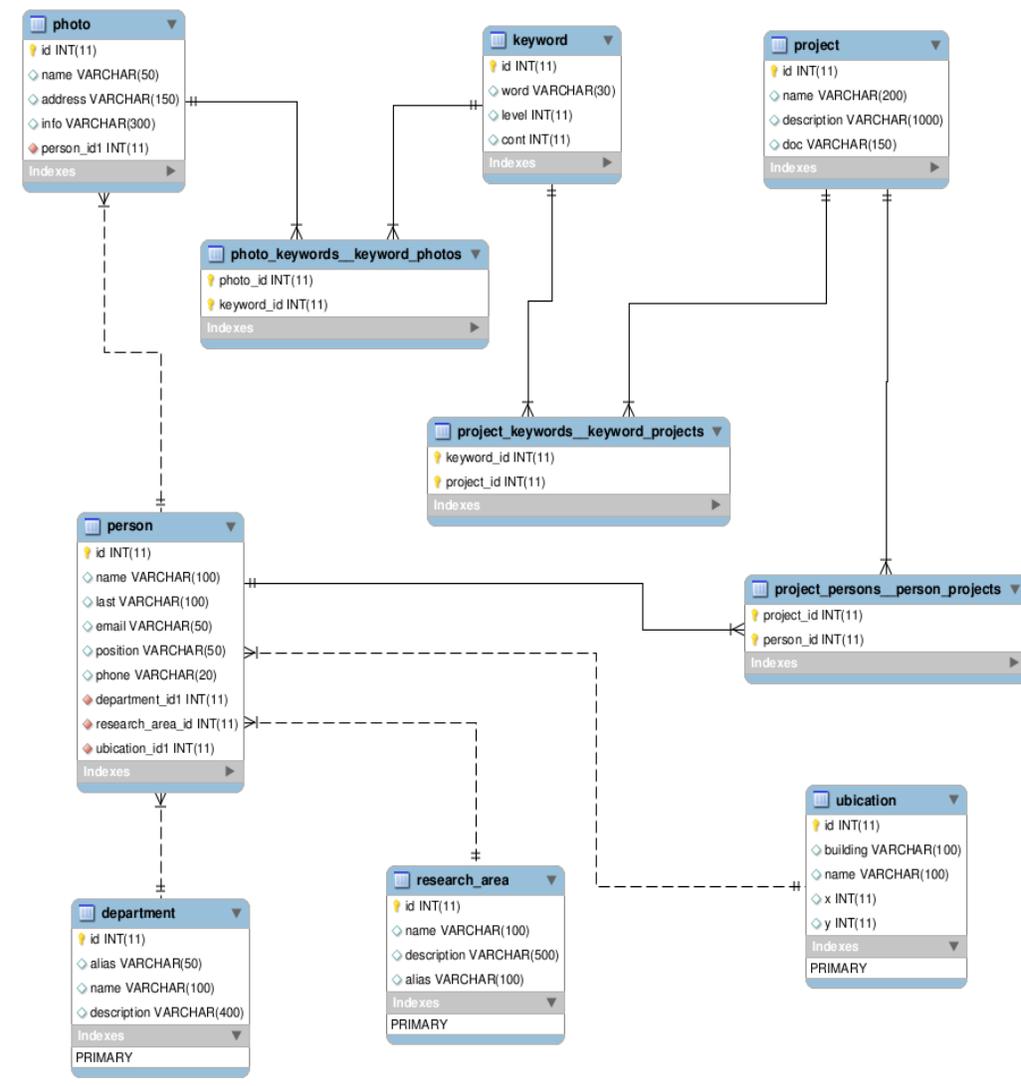


Figura 34. Diseño Estructural de la Base de Datos

3.6 DISEÑO DEL PLAN DE PRUEBAS

La mesa multitouch incorporará las siguientes pruebas: velocidad de respuesta ante una acción particular considerando la finalización de la tarea y usabilidad del Sistema. Para esta fase de pruebas se involucrará a 8 usuarios; el perfil de estos usuarios son estudiantes de ESPOL y personas con poco conocimiento de manejo de interfaces, a los que se les pedirá realizar 4 tareas en la mesa. Estas tareas relacionadas a la funcionalidad serán:

- Realizar la Búsqueda por medio de arrastrar palabras claves.
- Ver Detalles de una persona.
- Ver la ubicación en el mapa.
- Ver detalles de un proyecto en particular

En lo que respecta a usabilidad se buscará medir facilidad de uso de la aplicación por medio de la interacción con los dedos, a través de las mismas tareas planteadas para medir funcionalidad; ya que, uno de los objetivos de esta tesis fue implementar una aplicación sencilla de usar en un ambiente natural.

Con respecto a Funcionalidad se medirá la eficiencia de la aplicación por medio del tiempo que cada usuario se tome en realizar cada una de las tareas.

Adicionalmente a los tiempos de ejecución de tareas y facilidad de uso, también se realizaron anotaciones, para identificar donde se daban problemas o las razones por las cuales se les complicaba la ejecución de alguna tarea.

CAPITULO 4

4. IMPLEMENTACIÓN

En este capítulo se presentan los componentes usados tanto del lado del Cliente como del Servidor; además, se analizará como se implementó lo analizado y diseñado en el capítulo anterior, luego seguiremos con la revisión en detalle del porqué del uso de algunas herramientas que son parte de este Sistema.

La Figura 35 indica cada uno de los componentes usados para el Sistema y la forma cómo interactúan entre ellos. La implementación del Sistema se hizo tanto a nivel de Hardware, como a nivel de Software, en las siguientes secciones se presenta en detalle el proceso de desarrollo de esta solución basándonos en el Análisis y Diseño hecho previamente.

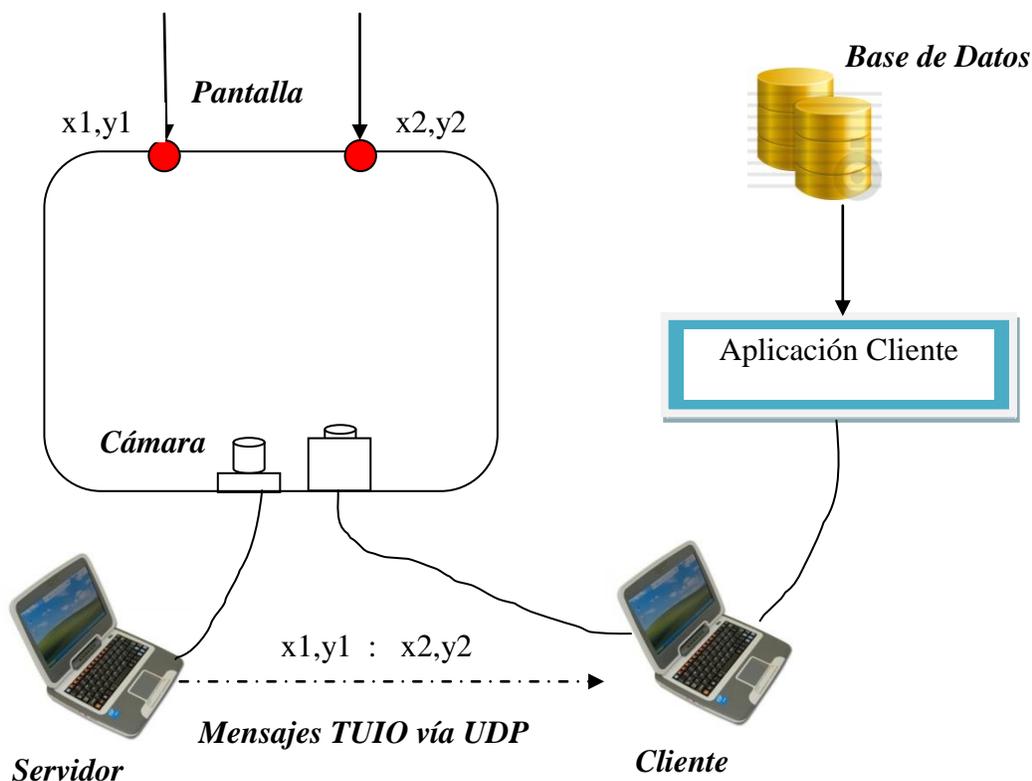


Figura 35. Esquema general del Proyecto.

Implementación del Hardware.-

Comenzaremos por describir la ubicación de cada componente. Las luces se las ubicó de tal forma que describan un arreglo de 3 x 4 como se describe en la Figura 36. El esquema de posición de las lámparas se lo hace para iluminar toda la superficie y que la luz infrarroja llegue a todos los puntos con igual intensidad.

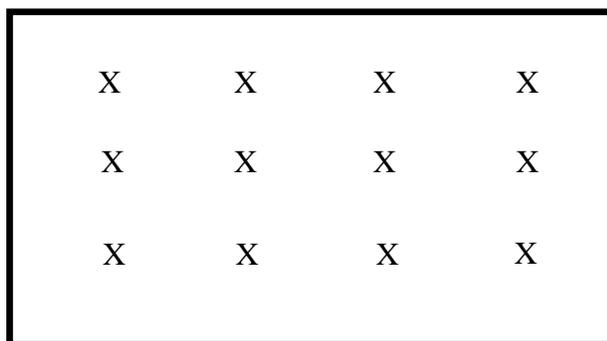


Figura 36. Arreglo de Luces.

La cámara se la ubicó en el centro de la base de la mesa, debido a que se requiere enfocar toda el área de la superficie. El proyector se ubicó en la base de la mesa y frente a él, se colocó un espejo para reflejar al lado contrario del proyecto (Ver esquema en la Figura 37).

En la Figura 38 podemos observar el posicionamiento de cada uno de los componentes internos de la Mesa Multitouch.

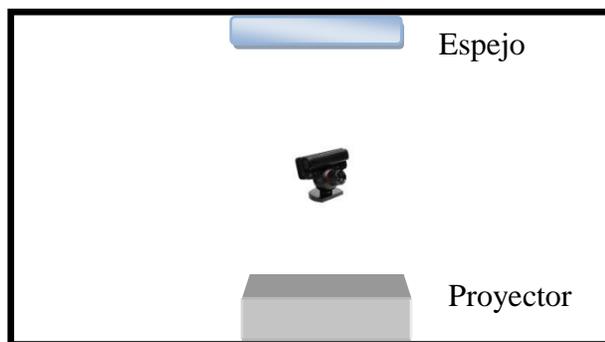


Figura 37. Posición del Proyector, Espejo y Cámara.

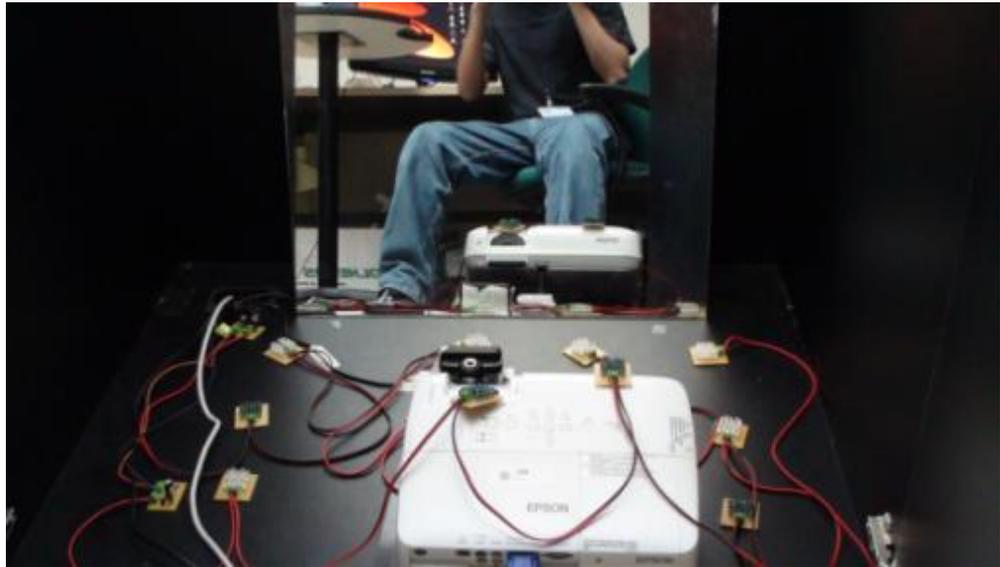


Figura 38. Vista interior de la Mesa Multitouch

Las lámparas se las ha colocado de tal forma que todas estén a la misma distancia unas de otras, las medidas se las puede observar en la Figura 39.

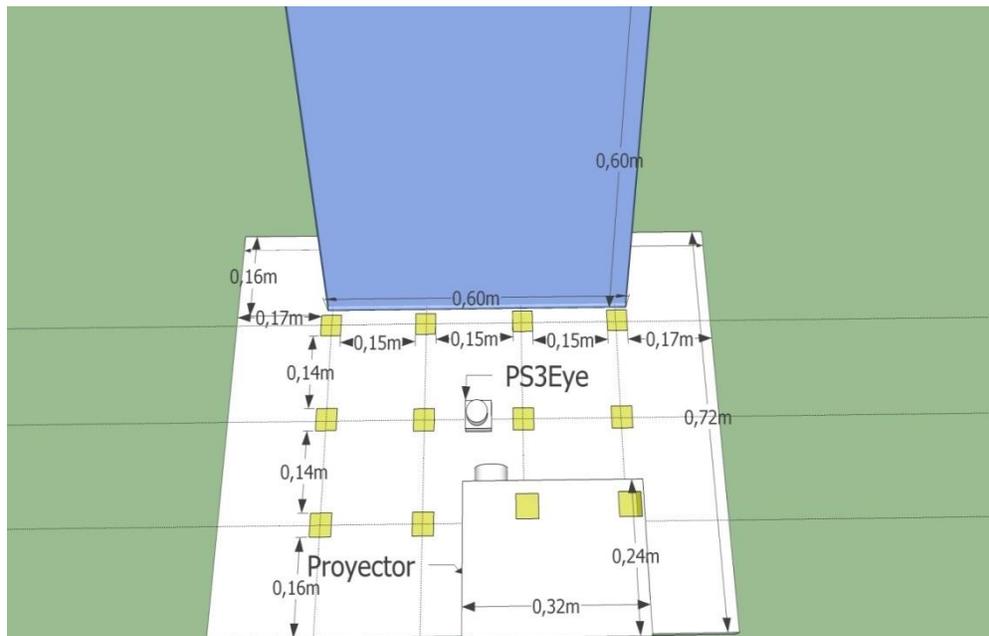


Figura 39. Medidas de los elementos internos de la mesa multitouch

Cómo se menciona en el capítulo de Análisis y Diseño, para la superficie se necesitó de un difusor, el cuál es un material especial de plástico que absorbe poca luz, en la Figura 40 podemos observar el exterior de la superficie multitouch.



Figura 40. Vista exterior de la Mesa Multitouch

Implementación del Software.-

Para llevar a cabo el desarrollo de cada uno de los módulos del software especificados en el capítulo 3, se hizo uso de las siguientes herramientas:

- Python 2.6.5
- SQLAlchemy 0.6.0 con Elixir
- PyMT 0.4
- MySQL

A continuación la justificación de la selección de estas herramientas en la implementación de este proyecto.

Python.-Este lenguaje de programación ofrece gran soporte para la integración con otros lenguajes y herramientas lo cual es muy bueno al momento del desarrollo bajo diferentes tipos de ambientes. Otra de las ventajas que tiene es que python puede ser aprendido en pocos días es decir su curva de aprendizaje es poco pronunciada.

SQLAlchemy 0.6.0 con Elixir.- SQLAlchemy es un Mapeador de objetos relacionales (ORM) que es un paquete de python para el mapeo de las Tablas a Clases; de tal forma que, se manejen los datos como objetos desde la base. Es un proyecto Open Source y cuenta con una gran comunidad de desarrollo. Tiene mucho que ver con patrones de persistencia, diseñados para eficiencia y alto rendimiento de los accesos a la base de datos, y lo mejor de todo es que su sintaxis es simple ya que está basado en Python.

Con SQLAlchemy tenemos todo el poder de las sentencias SQL por medio de comandos simples escritos en python. El principal motivo de

esta tesis para el uso de un ORM, es que la arquitectura es independiente de la Base de Datos que se use, de tal forma que el Sistema que se implementa, con el cambio de una sola línea puede ser ejecutado, usando un motor de base de datos diferente.

Este ORM cuenta con extensiones y plugins desarrollados por miembros de la comunidad, con el fin de mejorar ciertos procesos del ORM, uno de estos plugins es Elixir el cuál provee de una sintaxis sencilla al momento de establecer las relaciones entre las Tablas de una Base de Datos de tal forma que se reduce significativamente las líneas de código, además de lograr un código totalmente legible y entendible, fácil de administrar y actualizar.

La parte del modelo de nuestra aplicación está totalmente basada en el uso de SQLAlchemy y Elixir, la información con que se llena la Base de Datos se encuentra en un script que al ser ejecutado, realiza todas las transacciones sobre la Base de Datos. Para cada entidad de la base de datos contamos con su respectiva clase en nuestro modelo, cada una de las clases heredan de un objeto llamado Entity que cuenta con atributos y métodos para las transacciones en la Base.

MySQL.-Como motor de Base de Datos se ha elegido mysql debido a que es de código abierto y muy usada a nivel de pruebas y desarrollos iniciales, como se dijo en la sección previa, el manejo de los datos es independiente del motor que se use, si en algún momento se requiere usar un motor diferente simplemente se cambia la línea de conexión.

PyMT.-Es un framework especializado para la creación de aplicaciones multitouch, cuenta con componentes como botones, ventanas, formularios, entre otros. Es open source y multiplataforma. Soporta gran cantidad de dispositivos multitouch de entrada entre los cuales tenemos al **protocolo TUIO**, el cuál es uno de los principales motivos para elegir esta librería.

PyMT tiene detrás una gran comunidad open source como los de NUI Group, y su desarrollo es activo, cada versión mejora los rendimientos en velocidad y procesamiento de la anterior. El código fuente está disponible para ser descargado y usado, gracias a ello en caso de encontrarse un bug, se lo puede reportar para ser solucionado.

En el proceso de desarrollo se encontró un bug⁵ en la librería y se publicó la solución en el foro de desarrolladores, el bug tenía que ver con el uso de un cambio de estado en el componente MTToggleButton.

La capa de presentación está a cargo de vista.py que es donde se manejan toda clase de eventos y diagramación del layout.

El entorno de Desarrollo usado es Eclipse debido a que es open source, con la instalación de un plugin denominado pydev se crea un espacio de trabajo para programar en python.

⁵<http://code.google.com/p/pymt/issues/detail?id=294>

CAPITULO 5

5. PRUEBAS Y ANÁLISIS DE RESULTADOS

Para las pruebas de la pantalla multitouch con el Sistema de Búsqueda de Información se evaluaron a 8 personas (4 hombres y 4 mujeres) con una edad entre 22 y 32 años. Tres de ellos están muy familiarizados con computadoras.

Este capítulo corresponde al análisis de las pruebas realizadas, con la finalidad de evaluar el hardware en lo funcional y la usabilidad de la aplicación de búsqueda. Para ello se hicieron dos tipos de preguntas por cada tarea.

Las tareas están relacionadas directamente con la interacción sobre la pantalla multitouch, la primera tarea consiste en arrastrar 3 palabras al área de criterios de búsqueda, lo que se busca con esto es familiarizar al usuario con arrastrar y soltar objetos. La siguiente tarea consiste en seleccionar de la lista de Investigadores uno en particular para saber sus detalles principales, este ejercicio tiene como objetivo introducir al usuario en el uso de botones dentro de la aplicación. La tercera tarea implica ver detalles más específicos de la persona como su ubicación y los proyectos en los que ha trabajado, la finalidad de esta tarea es el

uso de gestos como agrandar, encoger y trasladar. La cuarta tarea involucra la lectura de un documento relacionado al proyecto, además de la consulta de las personas relacionados a éste, esta última tarea recopila actividades hechas en las 3 tareas previas.

Las pruebas se clasifican en dos tipos: de funcionalidad y de usabilidad. La funcionalidad se la medirá en base al éxito en la resolución de cada tarea, y la usabilidad en base al tiempo que duró cada tarea en realizarse, además de la dificultad que representó para cada usuario realizar las tareas.

Los 8 participantes en la prueba debieron seguir las instrucciones que se le entregaron en un pequeño documento que contenía las tareas y además un conjunto de preguntas relacionadas a las pruebas (ver anexo B)

5.1 PRUEBAS DE FUNCIONALIDAD

Estas pruebas ayudan a medir si un usuario logró hacer las tareas propuestas en la prueba con el uso de sus dedos. Para ello se realizaron 5 tareas(contabilizo 5 debido a que la tarea 3 tiene 2

subtareas), las cuales miden si el usuario consiguió realizar la tarea o no lo hizo.

En la Figura 41 observamos que solo el 3% de las personas, que fue 1 persona no pudo completar todas las tareas. Es decir de las 40 tareas en total que fueron asignadas a todos los usuarios solo hubo un usuario que falló en una tarea.

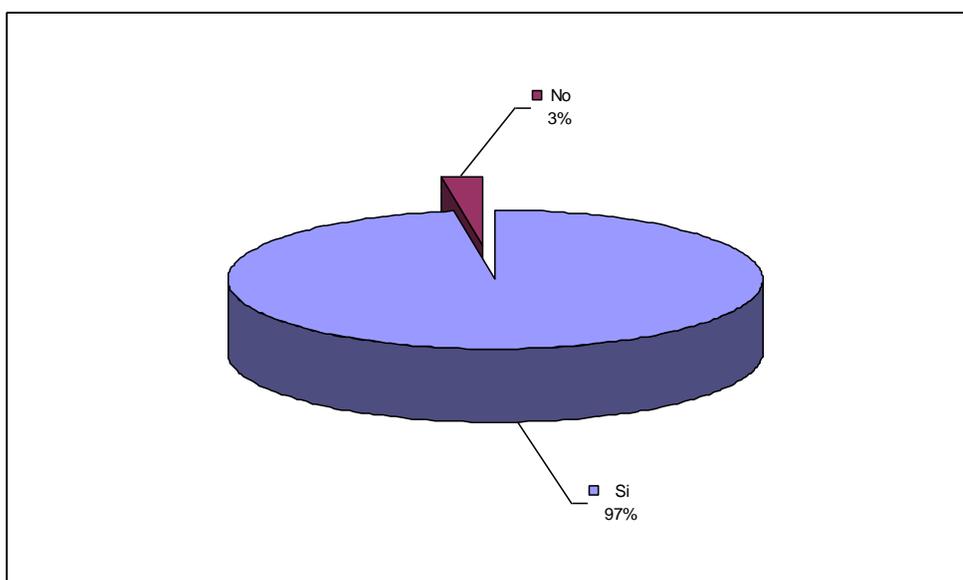


Figura 41. Gráfico de la Funcionalidad del Sistema respecto a la correctarealización de las tareas por parte de los usuarios.

Como podemos observar la gran mayoría de las tareas fueron respondidas de una forma afirmativa, es decir los usuarios pudieron llegar a terminar las tareas, lo cual indica que el Sistema trabaja correctamente en lo funcional.

En cada tarea se midió el tiempo empleado por los usuarios para completarla, 2 de los 8 usuarios requirieron leer dos veces la tarea para ejecutarla, específicamente en la segunda tarea 4 de 8 usuarios encontraron dificultades al momento de seleccionar a una persona para ver sus detalles, debido a que se hace uso de dos botones en vez de 1. En términos generales se logró el efecto que se buscaba, introducir a las personas a este nuevo tipo de interfaces y a su vez concluir cada tarea de una forma eficiente. Algunos usuarios necesitaron más tiempo por razones como: entender la orden que se les solicitaba en la prueba, por la poca familiaridad con interfaces multitouch, por problemas de precisión al presionar botones o arrastrar objetos, con el primer y segundo evaluador hubieron fallas de precisión con respecto a la mesa; esto se dio debido a un desfase en la calibración del tracker de 2 centímetros. De los 5 evaluados que no son expertos en computación, con uno se tuvo dificultades, e incluso recomendó un entrenamiento introductorio previo, en total se tuvieron 2 usuarios con problemas de precisión y 1 usuario con problemas de poca familiaridad. En el caso de los usuarios expertos en computación fueron los evaluadores que realizaron mayor cantidad de

movimientos y acciones sobre la pantalla, a diferencia de los usuarios no expertos en computación los cuáles se limitaron solo a completar las tareas y hacer lo que la ficha de observación ordenaba. En la Figura 42 observamos el tiempo promedio que cada usuario requirió para completar cada una de las tareas y la media general de finalización de cada tarea en segundos. Es decir la primera persona desde la izquierda en promedio cada tarea la concluyo en 59 segundos, la segunda en 38, etc.

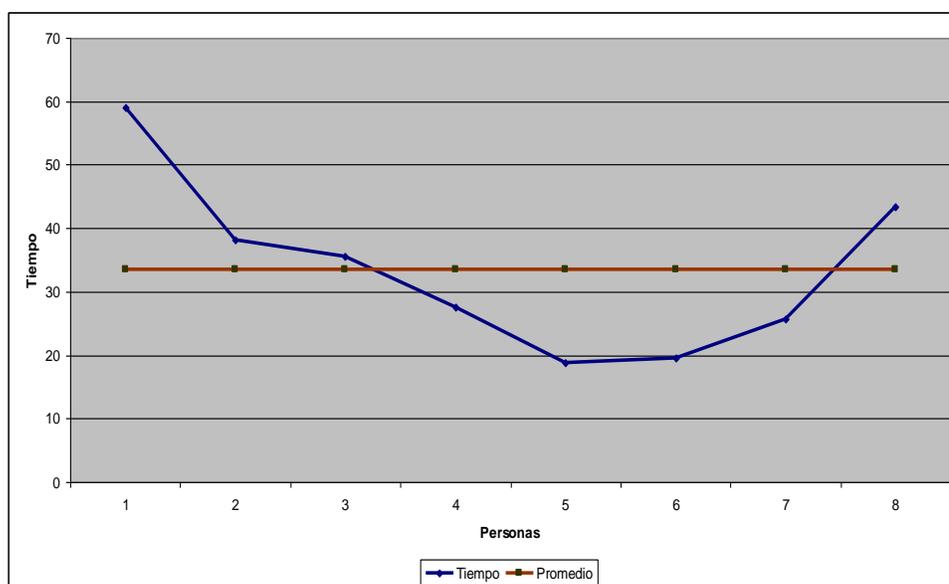


Figura 42. Gráfico del tiempo promedio que requirió cada usuario en completar cada tarea.

5.2 PRUEBAS DE USABILIDAD

Para medir la usabilidad del Sistema se evaluó la dificultad que representó para cada uno de los usuarios la realización de dicha tarea, a través de un documento que leían los participantes al ejecutar las tareas. Se utilizó una escala del 1 al 5, donde 1 es muy difícil y 5 muy fácil, para medir la facilidad de uso. En la Figura 43 podemos observar los promedios de dificultad presentados por tarea.

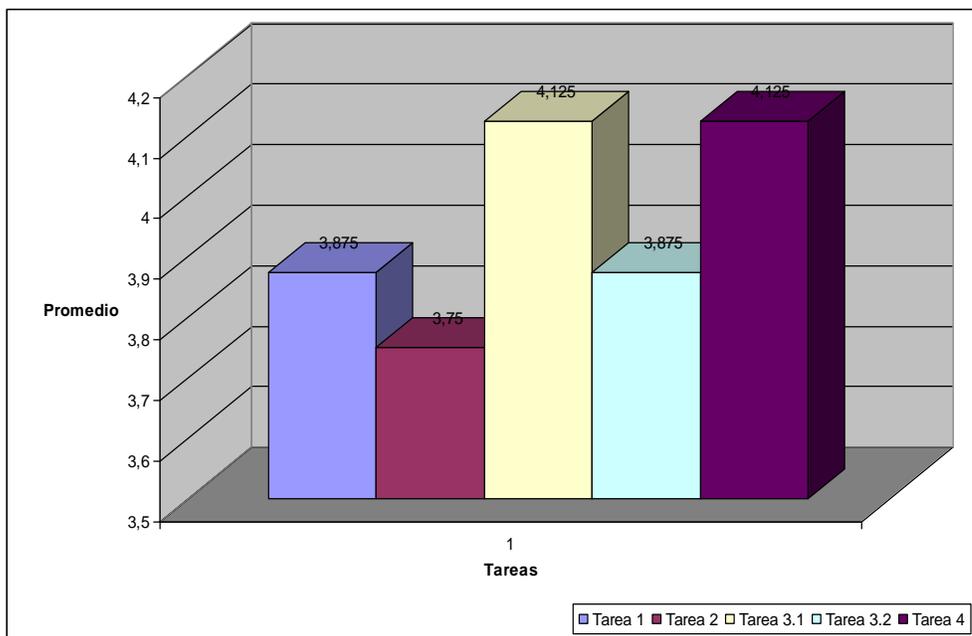


Figura 43. Gráfico de los promedios de dificultad por tarea.

Como se observa cada tarea tuvo un grado de dificultad diferente, el promedio de dificultad estuvo en 3.95/5, en la Figura 44 se aprecia de mejor forma las tareas que están sobre la media.

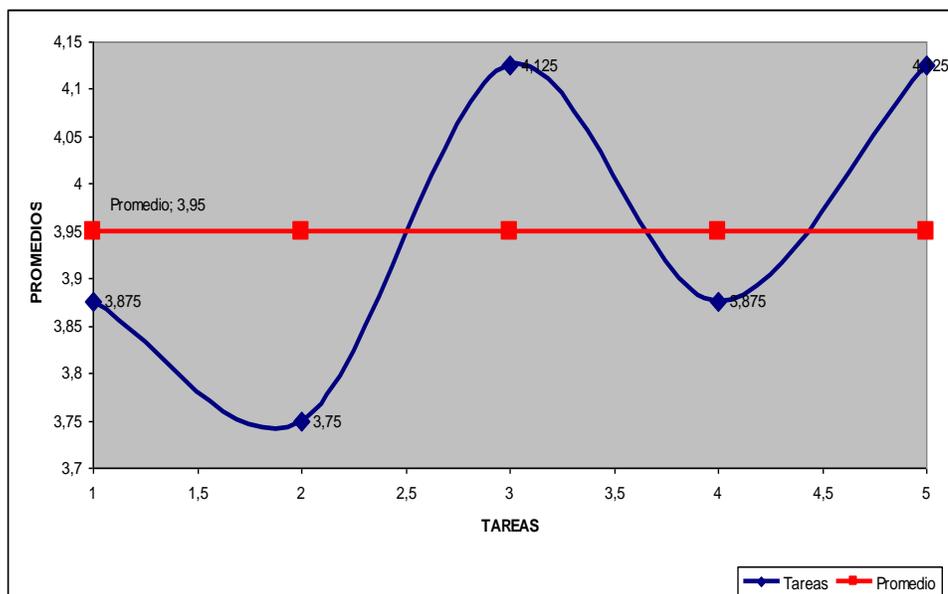


Figura 44. Gráfico que muestra promedio de dificultad por tareas y general

Las tareas 3 y 5 son las que se encuentran sobre la media, es decir que las tareas en las que tuvieron mayores dificultades fueron las que involucraron arrastrar objetos y presionar botones. Así mismo se observa que una de las tareas de menor dificultad fue la 5, dónde se presiona botones y se arrastran objetos, debido

a la experiencia que adquirieron en las tareas previas 1 , 2 , 3 , 4, al momento de la tarea 5 , les resulto fácil trabajar sobre algo que ya habían hecho previamente.

5.3ANÁLISIS DE RESULTADOS

Para las personas la facilidad de uso de la pantalla multitouch estuvo en el 79%, si bien no es un buen indicador, hay que considerar que se evaluaron a 3 personas expertas en computación y a 5 personas que no están inmersas en el mundo de la tecnología.

Considerando las características del hardware existieron problemas de detección en las esquinas debido a la modificación realizada sobre los lentes, que no son los originales de la ps3eye. Además de mayor sensibilidad en ciertas áreas de la pantalla, sin embargo esto no impidió la correcta realización de cada una de las tareas.

Completar cada tarea no significó un problema, solo una persona no pudo realizar la tarea3.2 que involucraba ver las personas

asociadas al proyecto, debido a que inconscientemente presionó un botón equivocado.

Los usuarios expertos en computación fueron más críticos en base a detalles como tamaño de botones, botones no visibles, diferencias entre botones o información en pantalla, etc. Algunas de sus sugerencias fueron: usar botones grandes, imágenes representativas en los botones como una flecha en el botón atrás, los nombres de los botones no son representativos, y por último disminuir la sensibilidad en la sección del mapa.

El affordance de la aplicación quedó expuesto principalmente en la transición de la tarea 2 a la tarea 3, cuando inconscientemente, se adelantaban en las tareas y presionaban el botón de maximizar para más detalles, cuando esa actividad era parte de la tarea 3. Los usuarios intuitivamente realizaban tareas que no necesariamente estaban expuestas en el documento que se les entregó para realizar la prueba, como las usadas en el mapa y la presentación de documentos.

Algo muy interesante de destacar fue que las personas con poco conocimiento de tecnología, se desarrollaron bien en el uso de la pantalla, aparte les resultó fácil y resolvieron las tareas propuestas en menor tiempo que las personas con conocimientos sólidos en computación. Principalmente esto se debe a que ellos se limitaban a cumplir lo que el documento entregado ordenaba.

Una peculiaridad destacable fue que la mayoría de personas evaluadas solo usaron una mano para manipular el sistema, solo uno de los evaluadores usó sus dos manos para escalar el mapa.

Como resultado de las pruebas, también se obtuvo opiniones de las personas evaluadas con respecto a este sistema, entre las que destacan una mejor visibilidad en los botones, mejorar las dimensiones del scrollbar en la lista de presentación de personas involucradas en los proyectos, con respecto al hardware un poco menos sensible al momento de manipular imágenes.

Se hicieron pruebas además de colaboración donde se tuvo hasta tres usuarios sobre la pantalla, jugando con una aplicación en la

cual el objetivo era no dejar pasar una pelota al otro lado y hacerla rebotar, además de jugar también tres en raya.

CONCLUSIONES

1. Se consiguió la implementación total de este Sistema a nivel de Hardware y Software con herramientas open source y sin ningún coste de Software. El hardware estuvo dentro de lo contemplado en el presupuesto y el precio del último prototipo, incluyendo solo materiales es de \$1289.54; es decir 10 veces menor que la opción de Microsoft Surface. por lo que, los objetivos propuestos inicialmente en este trabajo de graduación han sido cumplidos.
2. Los usuarios evaluados no mostraron rechazo al uso de este dispositivo, aún cuando para el 62% de ellos el uso de éste es realmente nuevo. Los resultados obtenidos dejan en claro que es necesario realizar ciertos cambios principalmente en el diseño de botones, y sensibilidad de la pantalla.
3. Se realizó el estudio de usabilidad y se analizaron aspectos a mejorar, entre los cuáles se destacan aspectos a nivel de interfaz, como por ejemplo el cambio de colores en botones, el uso de scrollbar en las listas, etc.

4. Se encontraron dificultades a la hora de ejecutar la tarea 2, en la cual al parecer hay un paso adicional no necesario, que involucra presionar dos botones para obtener detalles de una persona. Estas dificultades se vinculan con la falta de experiencia en el uso de estos dispositivos. La posible dificultad se puede minimizar suprimiendo uno de los botones, es decir el usuario escoja la persona que quiere ver directamente presionando sobre la imagen.

5. La dificultad en la manipulación de imágenes con el uso de los gestos, que se presentó en algunos de los evaluados, se debe a la poca experiencia que tienen con este tipo de dispositivos. La posible contrariedad encontrada se puede minimizar, logrando que el usuario se familiarice con este tipo de interfaces. Un ejemplo familiar puede ser al comienzo el uso de un iphone al momento de escribir un mensaje con el teclado en pantalla.

RECOMENDACIONES

1. Se recomienda el uso de una cámara firewire para tener mayor grado de precisión al momento de generar movimientos bruscos, además del uso de la mesa multitouch en lugares con poca iluminación o con iluminación no directa sobre la superficie.
2. Es recomendable usar otras herramientas diferentes a las usadas en este proyecto. Una de las herramientas que se debería analizar y que se podría usar en posibles implementaciones es MT4J que es un framework implementado en Java para la creación de aplicaciones multitouch.
3. Por último se debería realizar la calibración de la mesa en un lugar oscuro sin nada sobre la superficie; de forma que, los datos que se le den al tracker sean los más precisos.

ANEXOS

ANEXO A.- Descripción de cada una de las Tablas de la Base de Datos.

A continuación se detalla la función de cada tabla mostrada en el siguiente diccionario de datos, el cual hace referencia a las tablas de la Figura 18:

Tablas Primarias

- **Person.-** Tabla que almacena los datos básicos de las personas que son objetos de búsqueda en el Sistema, incluye nombres, apellidos, teléfono, correo electrónico y posición.
- **Department.-** Tabla que proporciona información de la Facultad a la que pertenece una persona. La información presente en esta tabla es alias, que es la abreviatura de la Facultad, nombre de la Facultad y Descripción.
- **Research_Area.-** Tabla que contiene información del campo de Investigación al cual la persona pertenece, los campos son alias, nombre y Descripción.
- **Ubication.-** Tabla que guarda los datos concernientes al Edificio y el nombre del Lugar donde la persona trabaja, además de coordenadas x y y que sirven de referencia de posicionamiento.

- **Photo.-** Tabla que contiene información necesaria para cargar una foto en el Sistema, datos como el nombre de la foto y la ruta para cargarla.
- **Keyword.-** Tabla que almacena las palabras claves, sus campos son la palabra, orden de importancia, un contador que sirve para almacenar la cantidad de veces que la palabra ha sido usada, y por último el campo info con una metadata de la foto.
- **Project.-** Tabla que proporciona información sobre los proyectos que han realizado las personas, esto incluye nombre, descripción y un path al documento.

Tablas de Relación

- **Photo_Keywords__Keyword_Photos.-** Tabla que sirve para establecer una relación entre las fotos y las palabras claves, para por medio de palabras claves encontrar fotos o viceversa.
- **Project_Keywords__Keyword_Projects.-** Tabla que asocia palabras claves a los proyectos, de tal forma que cada proyecto contiene un conjunto de palabras claves que lo identifica.

- **Project_Persons__Person_Projects.**- Tabla que define la relación existente entre la(s) persona(s) con su(s) proyecto(s) de Investigación.

ANEXO B.- Prueba de Usabilidad

Ficha de Observación Prueba de Usabilidad

Fecha: 07/09/2010

Lugar: Departamento del 3er. Piso CTI

Id. Participante: 00_

El objetivo de esta prueba es medir la influencia del hardware en el rendimiento del sistema completo. Además de analizar la dificultad que el usuario tiene al realizar tareas sobre una aplicación multitouch.

A continuación efectuará ciertas tareas luego de las cuales contestará a las preguntas presentadas en el siguiente cuestionario, para revisar que tan sencillo o complicado resultó realizarlas. En cada pregunta deberá colocar el grado de dificultad que representó para usted la tarea medida en una escala del 1 al 5, donde 1 representa mayor dificultad y 5 menor dificultad.

Cada Tarea será evaluada en base a la facilidad de uso y la precisión en la detección de sus dedos como dispositivo de entrada al Sistema.

Nota: Al iniciar cada tarea anote la hora de inicio y luego de contestar las preguntas coloque la hora de fin de la tarea.

TAREA 1

Hora de Inicio: _____ **Hora de Fin:**

Proceda a Arrastrar 3 palabras al área azul que se encuentra a su derecha.

¿Puede indicar si se le hace familiar este tipo de búsquedas?

[Si] [No]

Grado de dificultad (1 muy difícil, 5 muy fácil)

[1] [2] [3] [4] [5]

TAREA 2

Hora de Inicio: _____ Hora de Fin: _____

Proceda a Seleccionar una persona de la lista de Investigadores mostrados para ver más información de estos, tocando en Ver Detalles y luego en ver.

¿Podría decirme cuál es el nombre del Investigador y su email?

[Si] [No]

Grado de dificultad (1 muy difícil, 5 muy fácil)

[1] [2] [3] [4] [5]

TAREA 3

Hora de Inicio: _____ Hora de Fin: _____

Una vez que escogió a un investigador, busque dónde trabaja esta persona en ESPOL tocando el botón de maximizar en la ventana flotante.

¿Podría decirme si el programa le permite saber donde trabaja este investigador en el mapa de la ESPOL?

[Si] [No]

Grado de dificultad (1 muy difícil, 5 muy fácil)

[1] [2] [3] [4] [5]

Este mismo investigador tiene asociado proyectos en los que ha trabajado, indique cuáles son estos proyectos, lo pudo hacer?

[Si] [No]

Grado de dificultad (1 muy difícil, 5 muy fácil)

[1] [2] [3] [4] [5]

TAREA 4

Hora de Inicio: _____ Hora de Fin: _____

Proceda a seleccionar uno de los proyectos de la Lista de Proyectos mostrada a la izquierda de la ventana. ¿Puede decirme de que se trata y además las personas involucradas en él?

[Si] [No]

Grado de dificultad (1 muy difícil, 5 muy fácil)

[1] [2] [3] [4] [5]

Algún comentario adicional sobre el Sistema mostrado y la forma de interacción con el ¿facilidad de uso? ¿Algo destacable o no funcional?

BIBLIOGRAFÍA

[1] Inkpen, K., McGrenere, J., Booth, K. S., and Klawe, M. (1997). **“The effect of turn-taking protocols on children's learning in mouse-driven collaborative environments.”** In Proceedings of the Conference on Graphics interface '97 (Kelowna, British Columbia, Canada). W. A. Davis, M. Mantei, and R. V. Klassen, Eds. Canadian Information Processing Society, Toronto, Ont., Canada, pp.138-145.

[2]Kaltenbrunner, M., Bovermann, T., Bencina, R., Costanza, E.: **"TUIO - A Protocol for Table Based Tangible User Interfaces"**. Proceedings of the 6th International Workshop on Gesture in Human-Computer Interaction and Simulation (GW 2005), Vannes, France, 2005

[3]Kaltenbrunner, M., Bencina, R.: **"reactIVision: A Computer-Vision Framework for Table-Based Tangible Interaction"**. Proceedings of the first international conference on "Tangible and Embedded Interaction" (TEI07). Baton Rouge, Louisiana, 2007

[4]NUIGroup Authors: **“Multi-Touch Technologies”** 1st edition [Community Release]: May 2009, <http://nuicode.com/projects/wiki-book/files>

[5] OpenGL Architecture Review Board, Dave Shreiner, Mason Woo, Jackie Neider, Tom Davis. **OpenGL Programming Guide: The Official Guide to Learning OpenGL. (“The Red Book”)**. (Amazon link: <http://www.amazon.com/exec/obidos/ASIN/0321481003/>)

[6] Han, Jefferson Y. **“Low Cost Multi-Touch Sensing through Frustrated Total Internal Reflection.”** Symposium on User Interface Software and Technology: Proceedings of the 18th annual ACM symposium on User interface software and technology. Seattle,WA, USA, 2005. 115-118.

[7] Wright, M., Freed, A., Momeni A.: "**OpenSound Control: State of the Art 2003**". Proceedings of the 3rd Conference on New Instruments for Musical Expression (NIME 03), Montreal, Canada, 2003.

[8]**Touchlibhome**, NUIGROUP, <http://www.nuigroup.com/touchlib/>, 20/6/2010

[9]**Community Core Vision**, NUIGROUP, <http://ccv.nuigroup.com/>, 23/06/2010

[10] **Touch Screen – History of the Touch Screen Computer Interface**, About.com: Inventors, <http://inventors.about.com/library/inventors/bltouch.htm>, 27/06/2010

[11]**Extensible Markup Language (XML)**, W3C, <http://www.w3.org/XML/>, 27/06/2010

[12] **Apple – iPhone4 – Video Calls, multitasking, HD video, and more**, Apple, <http://www.apple.com/iphone/>, 10/09/2010

[13]**opensoundcontrol.org**, <http://opensoundcontrol.org/implementations>, 9/7/2010

[14] Septian H., Tao J., y Tan JP., "**People counting by Video Segmentation an Tracking**", IEEE Control, Automation, Robotics and Vision, 2006. ICARCV '06. 9th International Conference on, Dic. 2006.

[15]**TUIO implementations**, tuio.org, <http://tuio.org/?software> , 11/07/2010

[16]**Reactable-Products**, Reactable Systems, <http://www.reactable.com/products/>, 17/07/2010

[17]**MIDI Message Table 1**, MIDI Manufacturers Association, <http://www.midi.org/techspecs/midimessages.php>, 17/07/2010

[18] **Multitouch Interface nvision08**, Nvision 08,
<http://www.youtube.com/watch?v=EQFaBTba8qA>, 19/07/2010

[19] **Infotouch: An Explorative Multitouch Interface**, InfoTouch,
<http://www.youtube.com/watch?v=DHMJJwouq5l&feature=related>, 19/07/2010

[20] **Perceptive Pixel**, Perceptive Pixel,
<http://www.youtube.com/watch?v=ysEYwa-vHM>, 19/07/2010

[21] **Welcome to Microsoft Surface**, Microsoft,
<http://www.microsoft.com/surface/en/us/default.aspx>, 19/07/2010

[22] **Purchasing Microsoft Surface**, Microsoft Surface,
<http://www.microsoft.com/surface/en/us/Pages/HowToBuy/HowToBuy.aspx>,
19/07/2010

[23] **Multitouch G3 – PQ Labs**, PQ Labs, <http://multi-touch-screen.com/>,
19/07/2010

[24] **Movid**, Movid, <http://movid.org/>, 19/07/2010

[25] Lawrence Muller: “**Multi-touch displays: design, applications and performance evaluation**”. Grid Computing – Master’s Thesis.

[26] **Difference Between.net**, Difference Between.net,
<http://www.differencebetween.net/technology/difference-between-usb-1-1-and-2-0/>, 3/08/2010