

Capítulo 2

Primero se referencian los servicios de geolocalización, dando una explicación y después se enumeran proyectos relacionados con esto. Posteriormente se presenta la parte multicontacto igualmente explicando qué es y mencionando proyectos relacionados. Por último se enumera lo que se pueda relacionar con mejoras al software que beneficien al usuario; estas se pueden dividir en despliegue de información al usuario o en el funcionamiento interno del programa.

2.1 Geolocalización

El concepto de geolocalización consiste en encontrar la posición geográfica exacta de algún aparato o persona [Podnar et al. 2006]. Esta posición se representa por latitud y longitud en el globo terráqueo formando un punto que puede ser graficado en un mapa mostrando al usuario el lugar exacto. La latitud representa la altura a la que un punto se encuentra y la longitud representa cuántos espacios hacia los lados se quiere mover. De esta manera el punto 0,0 representa el centro de un planisferio, situado en la intersección entre el meridiano de Greenwich y el ecuador (Figura 2.1).



Figura 2.1-Coordenada 0,0 en un mapa. Captura de pantalla tomada de www.maps.google.com.

Actualmente hay algunas empresas que pueden manejar la geolocalización y brindan sus archivos al público mediante librerías. Dichas librerías tienen la capacidad de convertir una dirección en un punto en el mapa, es decir el usuario puede teclear en el campo de texto correspondiente la cadena de texto “México DF” y automáticamente el mapa se ubica en el centro del Distrito Federal; este proceso funciona también de manera inversa e incluso con direcciones específicas como la calle, número y colonia.

La empresa de mayor prestigio y al mismo tiempo la más solicitada que ofrece los servicios mencionados es Google, aunque también Yahoo! y otras cuentan con ellos, habiendo empresas como Nokia que quieren incursionar en

este campo por primera vez. En este trabajo se explican únicamente los servicios de Google, los cuales se hicieron disponibles para el público en febrero de 2005.

2.2 Aplicaciones de geolocalización

En el aspecto de geolocalización hay un buen número de proyectos enfocados a su uso. Bajo este contexto las personas se encuentran viendo un mismo punto para manipulación de esa imagen por parte de cualquier integrante. Esto en general es contraproducente para WITS ya que en la aplicación se proporciona una imagen distinta para cada usuario en vez de una imagen para todos, es por esto que la gran mayoría de trabajos sobre esta área no fueron tomados en cuenta.

Google provee además de la posibilidad de visualizar los mapas y las herramientas antes mencionadas una serie de controles ubicados generalmente del lado izquierdo de la pantalla que permiten alejar o acercar el mapa de manera rápida conocido en conjunto como Google Maps. Su interfaz cuenta con un cursor de navegación utilizado para desplazarse hacia el norte, oeste, sur y este sin problemas. Justo al lado del mapa se puede encontrar información de lugares que se encuentren cerca del punto en el que el usuario esté ubicado, como restaurantes, sitios de interés turístico, entre otros, aunque esto es opcional.

Google Maps cuenta también en sus librerías con un servicio que permite trazar la ruta más corta de un punto a otro y que respete las reglas de navegación, es decir, no trazará una línea recta de punto a punto sino que tomará las calles indicadas para llegar, calculando además la distancia que hay entre un punto y otro siguiendo la ruta mostrada. Los servicios de esta empresa dan la posibilidad

de colocar marcadores en el mapa como referencia a una locación específica que sea de utilidad para otras personas que visiten el mapa. No hay límite en cuanto a la cantidad de marcadores que se pueden colocar y además es posible cambiar su apariencia, aunque Google ofrece algunos marcadores predefinidos.

En el primer acercamiento de una aplicación de Google Maps con superficies táctiles fue usado un proceso que es capaz de establecer propiedades de multicontacto a cualquier aplicación [Olwal et al. 2008]. Uno de los ejemplos con los que mostraron su funcionalidad fue precisamente con Google Maps. Este proceso consiste en una serie de archivos que se encuentran funcionando en todo momento en la computadora, se instala en la computadora y permite que lo que se encuentre en la pantalla se convierta en objetos que puedan ser utilizados mediante el toque de los dedos, dando la posibilidad de acercar o alejar la imagen, rotarla entre otras cosas.

Un ataque al problema de este estilo no tendría éxito en la aplicación debido a que este proceso convierte las ventanas que están corriendo en ese momento a objetos multitáctiles, sin embargo no se puede llevar un control de los objetos o componentes que se encuentran dentro de la aplicación; en el caso de WITS esto se traduce a que no se podrían usar las zonas de control con este tipo de arquitectura, imposibilitando el uso de la aplicación.

Es productivo recurrir a la colocación de marcadores directamente en archivos HTML, estableciendo la longitud y latitud donde se quiere colocar de la misma forma que Sato [2008], lo que se puede llevar a cabo en WITS una vez sabiendo la posición exacta de los enemigos. Dicho proyecto difiere en que la información de los lugares donde se quiere establecer un marcador está

previamente establecida en una base de datos de la cual se recupera la información mediante métodos previamente definidos y consultas desplegando la información por medio de código PHP; el caso de WITS es diferente porque la posición del jugador es cambiante y se calcula al momento haciendo que el uso de bases de datos disminuya el ritmo de flujo de la información.

Street View es un servicio ofrecido por Google Maps, el cual cuenta con los mismos servicios de navegación, acercamiento y alejamiento pero con una perspectiva diferente, mostrando esto desde la perspectiva de un transeúnte caminando en medio de la calle. Este servicio funciona desde mayo de 2007 pero a principios del año 2009 se extendió el servicio a un número mayor de ciudades en el mundo.

Al igual que Google Maps, Street View cuenta con su catálogo de librerías que permiten a los programadores hacer uso de sus servicios sin costo alguno, lo cual permite el manejo de eventos y “escuchadores” que pueden a su vez disparar ciertas acciones asociadas al mapa o algún componente de la interfaz. Incluso posee una opción de panorama que hace que en un punto dado y sin detenerse el usuario pueda girar sobre su propio eje teniendo una vista de trescientos sesenta grados del lugar.

En cuanto al servicio de Street View no hay aplicaciones multicontacto conocidas, incluso no se han encontrado aplicaciones normales que utilicen este servicio en la biblioteca de ACM. Actualmente Street View posee varias características importantes que permiten su correcto funcionamiento, las cuales se explican a continuación.

Al usar Street View en el sistema, parte fundamental del mismo es el

sistema de navegación. Con el objetivo de explicarlo de forma plena es importante aclarar que las librerías de Street View establecen algunos parámetros que dan cabida a la orientación y localización en las calles, los cuales son explicados en esta sección para después analizar el sistema propuesto de navegación aplicado a las tecnologías multicontacto.

2.2.1 Elementos necesarios para controlar la localización

La localización de un usuario en el mapa es posible gracias a cinco elementos principales, latitud, longitud, orientación, ángulo de elevación y grado de acercamiento. Los primeros dos elementos constituyen el grupo de las coordenadas, el resto conforman el grupo de “punto de vista”. La única limitante en cuanto al movimiento son las calles que no tienen integrado el servicio.

El grupo de las coordenadas está conformado por latitud y la longitud; la latitud consiste en el lugar correspondiente en el eje “y”. Análogamente la longitud indica la distancia en el eje “x”.

2.2.2 Punto de vista

Una vez que se tienen las coordenadas indicadas se determina la posición en el planeta Tierra en la que se encuentre el usuario, quedando por especificar el lugar hacia el que se quiere mirar. Esto es posible gracias al elemento llamado punto de vista, que consiste de tres parámetros distintos, orientación, ángulo de elevación y grado de acercamiento.

La orientación se refiere al punto cardinal hacia el cual se está mirando, el cual es medido en grados, siendo cero cuando se está observando exactamente hacia el Norte. Los grados aumentan hacia la derecha de tal forma que el Oeste se encuentra en doscientos setenta grados, el Sur en ciento ochenta grados y el

Este noventa, teniendo la posibilidad de estar en un lugar en el mundo y poder ver lo que se encuentra alrededor.

El ángulo de elevación se refiere al lugar hacia donde se dirige la mirada verticalmente. El ángulo cero se refiere a la cámara mirando horizontalmente, si el usuario quiere voltear hacia el cenit sería necesario colocar este ángulo en un valor de noventa y si se quiere mirar hacia el piso debe colocarse en un grado de menos noventa, sin embargo esto último no es posible ya que el eje de la cámara no permite girar tanto sobre su propio eje “y” hacia abajo. El ángulo de elevación no es fundamental para el sistema, sin embargo al ser parte del punto de vista es necesario saber controlarlo para obtener resultados específicos.

El nivel de acercamiento define qué tan próxima se encuentra la cámara hacia un objetivo particular. Esta característica está disponible para el asfalto y los edificios que se encuentran a los lados de las calles, dando a este elemento una relevancia nula para WITS.

Street View, posee un cliente encargado de analizar una dirección sin llevar al usuario a ella y regresar la información de la misma, entre los datos que regresa este cliente se encuentra un valor booleano que establece si es una dirección válida o no. Adicionalmente, este cliente, cuyo nombre es GStreetViewClient, tiene la capacidad de que en caso que las coordenadas provistas no sean válidas puede indicar a la persona que requirió la información la latitud y longitud de la dirección válida más cercana y opcionalmente llevar al usuario a esta dirección. Esto lo realiza automáticamente.

2.3 Tecnología multicontacto

La tecnología multicontacto tiene tiempo en el mercado pero ha cobrado importancia recientemente; sin embargo, esto es a un nivel comercial donde existen aplicaciones para empresas privadas donde un usuario típico no tiene alcance. Este apogeo surge debido a la fabricación de superficies que aplican esta tecnología de manera eficaz [Kin et al. 2009] como Diamond Touch y Microsoft Surface. Esta tecnología se aplica a ciertas superficies como mesas, pantallas y pizarrones de manera que hay varios dispositivos que permiten el uso de esta tecnología, cada uno con sus ventajas y sus desventajas. Las interfaces multicontacto permiten al usuario una interacción adecuada debido a que se asemejan a una forma natural de interacción humana, utilizando las manos sin necesidad de herramientas adicionales como lo son el ratón o teclado. Incluso hay investigadores que colocan reconocimiento de voz [Piper et al. 2008] en estas superficies eliminando incluso al teclado. Al mejorar la interacción con la computadora se ve reflejado directamente en el término de tareas con mayor velocidad [Kin et al. 2009] dejando satisfecho al usuario. Una característica importante de superficies de este estilo es, como su nombre lo indica, que pueden procesar al mismo tiempo una gran cantidad de puntos de contacto dando pie a aplicaciones que pueden ser usadas por varias personas en una sola sesión.

Actualmente las superficies de este estilo no se encuentran a menudo en nuestro alrededor debido a su alto precio; sin embargo se espera que en el futuro sea posible para usuarios comunes, es decir personas con un nivel socioeconómico medio o mayor el acceso a este tipo de tecnología en sus hogares como una herramienta común sustituyendo a la computadora. Últimamente se ha combinado esta tecnología con la de RFID [Valli y Linari, 2008]

permitiendo que la computadora distinga entre objetos que se encuentren en contacto con la superficie. Existen algunas con capacidad suficiente para detectar la forma de los objetos colocados sobre su superficie como puede ser la palma de la mano [Dang et al. 2009], un lápiz o cualquier otro objeto de uso común [Izadi et al. 2009].

En cuanto a las herramientas de desarrollo existentes se encuentran algunos ambientes de desarrollo integrado que proporcionan soporte en cuanto a codificación, como Qt, sin embargo no hay un estándar de programación. El evento de mayor importancia enfocado a esta área es “Interactive Tabletop Surfaces”, un congreso anual en el que se muestran los adelantos que se han hecho alrededor del mundo en cuanto a superficies multicontacto se refiere.

2.4 Aplicaciones multicontacto

Cada superficie multicontacto necesita de un controlador que sea capaz de determinar el lugar en la pantalla que está siendo tocado. Estos son llamados rastreadores y hay gran variedad de ellos. Las últimas versiones son capaces de detectar la forma del objeto que está haciendo presión en la pantalla y trabajar con esa forma. Existen rastreadores avanzados que son capaces de detectar el grado de presión que se está ejerciendo. Desde luego no todos los rastreadores cuentan con estas características de detección y tener una de ellas no implica que por ello deberá tener las demás.

Estas librerías funcionan a base de detección de eventos sobre un área específica de la pantalla, definida por una clase llamada *Widget*, la cual define la interfaz del sistema, con características similares a las de una ventana pero que

además de mostrar al usuario los menús, opciones y el desenvolvimiento de la aplicación se encarga de recibir los eventos multicontacto. El *Widget* además puede detectar cualquier tipo de evento previamente establecido, como por ejemplo los movimientos del ratón, sin embargo WITS se enfoca únicamente a los sucesos multicontacto.

2.4.1 Características de detección

Las características similares que se han presentado en varias superficies táctiles de mesa, sean multicontacto o no, han dado pie a varios intentos de realizar una síntesis de ellas de manera que la programación sea mínima; uno de ellos es un protocolo multicontacto capaz de detectar cualquier evento de esta índole llamada TUIO con el poder de detectar varias características multicontacto sobre un área específica de la pantalla, definida por una clase que describe la interfaz del sistema, con características similares a las de una ventana pero que además de mostrar al usuario los menús, opciones y el desenvolvimiento de la aplicación se encarga de recibir los eventos multicontacto

Cada pixel del monitor es una unidad en la pantalla hacia cualquier dimensión. Es decir, existe un eje "x" y un eje "y". De acuerdo a estos dos ejes el origen se encuentra en la esquina superior izquierda del monitor. En el eje "x" entre más hacia la derecha se encuentre el pixel aumenta su valor, mientras que en el otro eje la posición es cada vez mayor conforme más hacia abajo se sitúe. TUIO no sólo detecta la posición en la pantalla sino la posición relativa al elemento que está siendo alterado, en cuyo caso posee también su propio eje coordenado con las mismas características que la pantalla.

Cuando se toca la superficie por primera vez, el punto exacto donde fue

realizado recibe el nombre de punto primario; automáticamente las librerías se encargan de almacenarlo en memoria para futuras referencias hasta que se despegar el dedo de la superficie indicando el término del evento detectado. Una vez finalizado el evento de inmediato se borran los valores en memoria del punto primario, permitiendo que el siguiente contacto en la superficie se convierta en el nuevo punto primario.

Al obtener el punto primario se activa el evento de punto de contacto inicial, al ocurrir este pueden suceder otros dos tipos de evento sobre él, los cuales son opcionales, dependiendo del usuario: actualización del punto de contacto y finalización del evento multicontacto (Figura 2.2).

Para actualizar un punto de contacto se necesita mínimo un punto de contacto inicial, de otra forma el evento de actualización del punto no se puede activar. Esta actualización se encarga de registrar los movimientos del punto primario siendo capaz de deliberar si está siendo presionado, si no está siendo ocupado, si se ha movido de lugar o incluso si no se ha desplazado indicando al sistema qué debe proceder.

La combinación de esta información con la última posición reconocida hace posible saber si el usuario movió su dedo hacia algún lado, lo cual con una simple evaluación de mayor o menor que se puede saber si fue hacia la izquierda o hacia la derecha.

En el momento que el usuario retira el dedo de la superficie se registra un punto final de contacto detectado por el evento de finalización de punto de contacto. Si el punto primario no se desplazó de lugar, el punto final será el mismo que el primario. Se puede dar el caso que el punto primario se haya desplazado,

pero al final regresó a su posición inicial, de manera que nuevamente sería el mismo al punto final.



Figura 2.2-Diagrama de flujo de puntos de contacto.

Aunque en general a cualquier clase se le puede agregar las propiedades que ofrece la librería multicontacto, pero hay una de particular interés. Como se mencionó anteriormente hay una clase que despliega la información al usuario y es también con lo que el usuario interactúa, funcionando como una interfaz. Dentro de esta hay varios objetos, los cuales también deben de tener incluidas las librerías multicontacto y que cambian su estado y apariencia al ser modificados.

2.4.2 Nori Scrum

El manejo de información que utilizaron en este proyecto [Voss y Schneider, 2009] parece indicado debido a que permite que el centro de la pantalla permanezca despejado. Este proyecto promueve mayormente el aprendizaje colaborativo. La diferencia en este es que utiliza unos pequeños paneles para cada usuario de tal forma que el usuario no tiene que realizar actividades en el centro de la pantalla evitando que estire el brazo hacia el centro de la pantalla estorbando la vista a los demás usuarios. Estos paneles cambian de posición a deseo del usuario, además de que cuentan con varios botones en su haber que realizan diferentes funciones al ser presionados, como mostrar un menú o teclado en pantalla justo debajo del panel.

2.4.3 Proyecto PAC PAC

Otro de los trabajos que tienen relación es un proyecto en el que se desarrolló una aplicación que no es totalmente multicontacto per se debido a que en ningún momento los jugadores tocan la pantalla, sin embargo se recurre también a los disparos [Sato et al. 2009], ya sea para eliminar al oponente o enviar objetos por medio de disparos hacia el área del oponente.

Evidentemente no hay mucha conexión entre el tipo de disparos que las dos aplicaciones tendrán; sin embargo, el concepto de eliminar a el o los oponentes para conseguir un fin sigue siendo el mismo, conseguir la posición del adversario y verificar si el proyectil lanzado dio en el blanco.

2.4.4 Propuesta de arquitectura.

Se comentó con anterioridad sobre la problemática que presenta el hecho de no

contar con un estándar de arquitectura establecido, sin embargo se ha propuesto uno que es conveniente para el desarrollo de este trabajo [Klompaker y Reimann, 2009] (Figura 2.3).

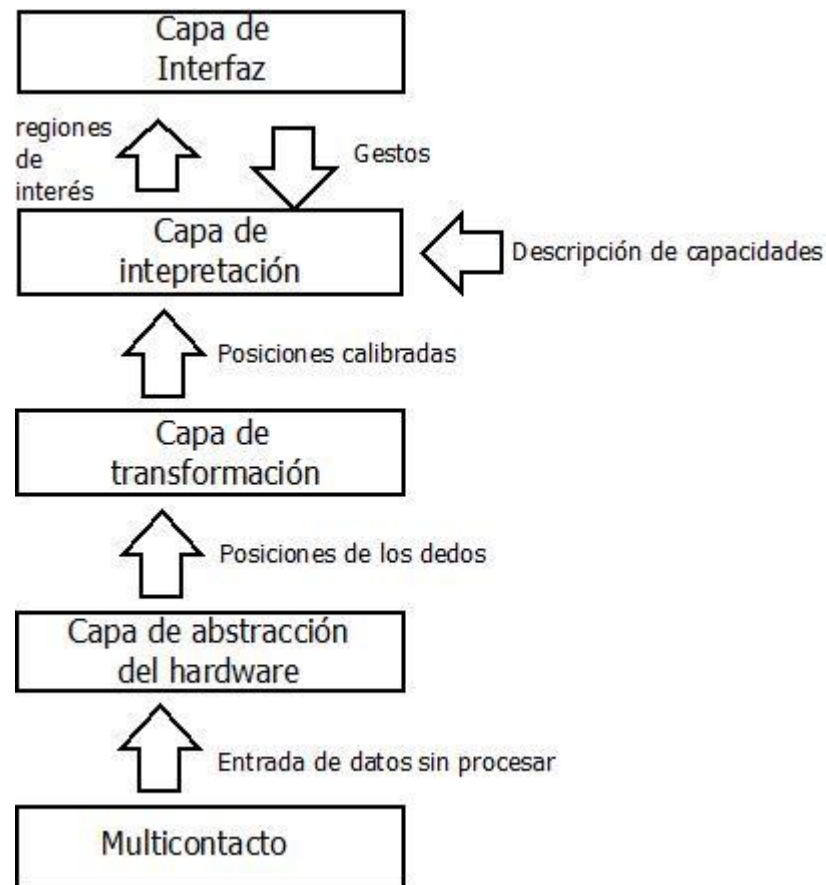


Figura 2.3-Arquitectura posible de multicontacto.

Empezando de abajo hacia arriba, lo primero que se ve es la superficie multicontacto, la cual detecta los puntos de presión y mediante un sistema puede regresar la posición exacta en que se realizó el contacto, la cual se refina en la capa siguiente. La capa de interpretación es la de mayor importancia debido a que aquí es donde se decide qué hacer con la posición de los dedos dependiendo de gestos. Entiéndase por gesto aquellos movimientos de las manos que guardan un significado para la computadora, por ejemplo tocar la pantalla con dos dedos y

después juntarlos en un mismo punto.

La región de interés es otro elemento a considerar, es decir, a qué parte de la pantalla y por ende los componentes que se encuentran dentro de esta parte afecta los gestos que el usuario realizó en un momento determinado. Una vez sabiendo el gesto y la región se describe lo que debe de realizar dando retroalimentación al usuario en una interfaz gráfica. Posteriormente inserta nuevos datos y el proceso de la capa de interpretación se vuelve a repetir sucesivamente hasta que la aplicación termine o el usuario deje de usarla.

La arquitectura presentada describe varios puntos que deben ser considerados: primeramente la capa de interpretación debe ser separada de la interfaz y trabajar independientemente cada una de ellas. Esto tiene varias consecuencias, primero que el código de cada una de ellas debe encontrarse en archivos separados, como ocurre en el paradigma MVC utilizado para aplicaciones web, proveyendo claridad al código y satisfaciendo uno de los objetivos de la aplicación.

Varias de las ventajas de esta arquitectura radican en que en general las cosas que se encuentran en las capas del fondo se encuentran previamente establecidas y no es necesario preocuparse por ellas. Esto da la oportunidad de enfocarse directamente a las capas de interfaz e interpretación a las cuales WITS se puede acoplar.

2.5 Otros beneficios a la aplicación en trabajos anteriores

En esta sección se enumeran algunos avances que ha habido en diferentes campos que no pueden ser catalogados específicamente como de tecnología

multitáctil y tampoco de geolocalización pero que pueden ser útiles para el desarrollo de la aplicación.

2.5.1 Posiciones de la mano.

En un estudio realizado se observó que en las superficies multicontacto el gesto de mayor utilización es el de apuntar a las cosas con el dedo índice [Olwal et al. 2008], el segundo en la escala es el de recargar sobre la superficie la palma de la mano con los dedos ligeramente separados.

En este estudio dicha posición de la mano es usada la mayoría de las veces para rotar objetos en la pantalla, por lo que se asemeja al control de una nave que va a dar vuelta a la izquierda o a la derecha. Además esto establece que dicho gesto es natural para el usuario y no le cuesta trabajo inferir este movimiento ni es una posición incómoda para el usuario, proveyendo una interfaz natural.

2.5.2 Colocación de enemigos.

En cuanto al funcionamiento del juego existen algunas estrategias como el colocar puntos de referencia [Voss y Schneider, 2009] para cada usuario dentro de un mapa en puntos distintos donde no haya un conflicto directo entre ellos sino hasta después de avanzado el juego.

Este mecanismo implica un algoritmo matemático entre zonas neutrales, es decir lugares donde el nuevo jugador puede aparecer sin peligro como el usado en el juego "Almansur Battlegrounds" basado en unidades hexagonales [Pereira et al. 2009].

2.5.3 Simulador

TUIO posee un simulador, el cual se encuentra disponible para su descarga.

Consiste de una ventana que no muestra nada sino un fondo blanco, la cual pretende que el clic del ratón sobre ella represente el toque de un dedo para de esta forma saber cómo reaccionará la pantalla a cada toque sin necesidad de utilizar una superficie multicontacto física. Utilizando una combinación del teclado y el ratón este simulador puede recibir varios puntos de contacto, únicamente al mantener presionada la tecla “control” y haciendo clic en alguna parte del simulador, lo que representa un dedo presionando la pantalla.

La ventana que se muestra representa la pantalla de la computadora pero en una escala menor, es decir, si se hace clic en la esquina superior derecha, es equivalente a un clic en la esquina superior derecha de lo que se encuentre en ese momento detrás de la ventana del simulador. Su funcionamiento es a base de conexión entre puertos de la computadora que facilita la comunicación entre la ventana del simulador y el programa que cuenta con la librería. Para lograr esto es necesario agregar algunas líneas de código en la aplicación. Hay un hilo que se encuentra monitoreando el paso de información entre los puertos que se inicializa al momento de arrancar la aplicación que se conecta al simulador.

2.7 Resumen del capítulo.

En este capítulo se analizaron algunos proyectos y avances que están relacionados a WITS, ya sea en la parte de la geolocalización, multicontacto o algunas ideas a desarrollar. En el capítulo siguiente se verá la arquitectura del sistema además de los algoritmos propuestos. Después se verá la implementación de los algoritmos en la aplicación para continuar con la evaluación y conclusiones del sistema.