

iTrackEye: Aplicación iOS para iPad basada en el tratamiento de los datos resultantes de Eye Tracking

José David Patón Romero ¹, Manuel Ortega Cantero ², Ana Isabel Molina Díaz ³, Miguel Ángel Redondo Duque ⁴

1) Escuela Superior de Informática, UCLM. Paseo de la Universidad, 4. 13071, Ciudad Real, Spain

JDavid.Paton@uclm.es

2) Escuela Superior de Informática, UCLM. Paseo de la Universidad, 4. 13071, Ciudad Real, Spain

Manuel.Ortega@uclm.es

3) Escuela Superior de Informática, UCLM. Paseo de la Universidad, 4. 13071, Ciudad Real, Spain

AnaIsabel.Molina@uclm.es

4) Escuela Superior de Informática, UCLM. Paseo de la Universidad, 4. 13071, Ciudad Real, Spain

Miguel.Redondo@uclm.es

Resumen

El desarrollo de aplicaciones para dispositivos móviles se encuentra en pleno auge debido al gran y creciente número de usuarios de “smartphones” y “tablets” que demandan este tipo de tecnología. Es por esta razón por la que se ha optado por el desarrollo de una aplicación de este tipo, enfocada en el ámbito de la investigación de la usabilidad.

El presente documento describe el proceso llevado a cabo para el desarrollo de la aplicación *iTrackEye* para el dispositivo móvil iPad de Apple. Esta aplicación utilizará la tecnología propia del iPad y de su sistema operativo actual (iOS 8) para representar gráficamente los resultados que se obtienen tras una sesión de *eye tracking*; mejorando de esta manera la experiencia del usuario gracias a la facilidad de interacción y ayudando a la mejora de la usabilidad y comodidad a la hora de realizar investigaciones en el campo que nos ocupa.

Así pues, *iTrackEye* permitirá la representación de los resultados obtenidos mediante mapas de calor, mapas de puntos ciegos o rutas de escaneo, basándose en cada uno de los participantes de cada test de un proyecto determinado de *eye tracking*.

Palabras clave: *Eye tracking*, *Eye tracker*, Seguimiento ocular, Mapa de calor, Mapa de puntos ciegos, Ruta de escaneo, Tobii Studio, iOS, OS X, iPad, Xcode, Objective-C.

1. Introducción

1.1. Planteamiento del Problema

En el ámbito del estudio de la usabilidad se llevan a cabo diferentes tipos de técnicas con el fin de aprender cómo un usuario interactúa, en este caso, con un determinado sistema informático, para intentar mejorarlo y adaptarlo a las nuevas necesidades que aparecen cada día.

Una de las técnicas más utilizadas son los test de usuario de pensamiento en voz alta (*thinking aloud*). En este tipo de técnica se le presenta una determinada tarea a un participante y se le insta a pensar en voz alta durante el desarrollo de la misma, indicando en todo momento qué es lo que opina. Esta técnica tradicional es muy útil y ha servido durante mucho tiempo para detectar la mayoría de los problemas de usabilidad de un sistema.

Una vez que se han detectado uno o varios problemas de usabilidad, un experto en el campo puede analizar cuáles son las causas del problema y proponer una serie de soluciones.

Sin embargo, esta técnica de *thinking aloud* también tiene sus inconvenientes, ya que, muchas veces, el comportamiento y los comentarios del usuario suelen ser ambiguos, poco concretos o incluso diferir en ciertos aspectos de lo que realmente piensa.

Por ello, a lo largo del tiempo se han ido creando nuevas técnicas y han ido apareciendo nuevas tecnologías que han permitido conseguir resultados más objetivos de estos test. Una de estas nuevas técnicas es el *eye tracking* o seguimiento ocular.

Mediante la técnica de *eye tracking* lo que se pretende es monitorizar y registrar la manera en que una persona explora visualmente una determinada escena, para determinar en qué áreas, durante cuánto tiempo y en qué orden fija su atención.

Para detectar todo esto se utiliza un *eye tracker*, que se trata de un dispositivo especial que lanza rayos infrarrojos a los ojos del usuario, que rebotan en las pupilas de éste y vuelven al dispositivo, lo que permite calcular con gran precisión el punto dónde dicho usuario mira.

Esta técnica de *eye tracking* no solo se utiliza en el ámbito de la mejora de la usabilidad, sino que el ámbito de aplicación alcanza a una gran variedad de áreas y disciplinas, como son el marketing y la publicidad, la investigación médica, etc.

Para posibilitar la interpretación de la gran cantidad de datos que se recogen durante una sesión de *eye tracking*, existen programas informáticos que generan una serie de animaciones y representaciones con el fin de resumir gráficamente el comportamiento visual de un usuario.

Estas representaciones se suelen recoger, principalmente, mediante mapas de calor (Figura 1), mapas de puntos ciegos (Figura 2) o rutas de escaneo (Figura 3).

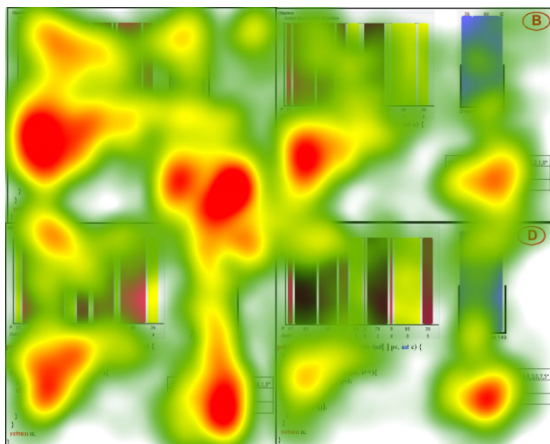


Figura 1 – Mapa de calor.

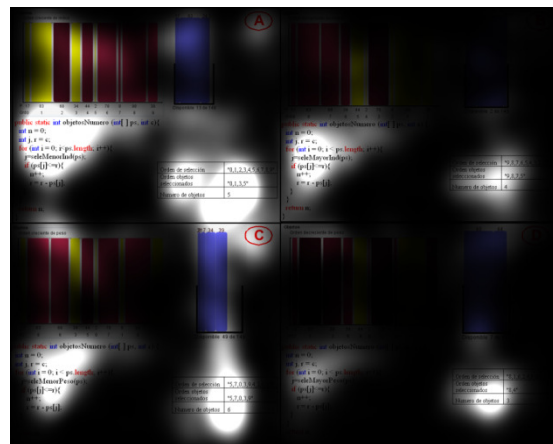


Figura 2 – Mapa de puntos ciegos.

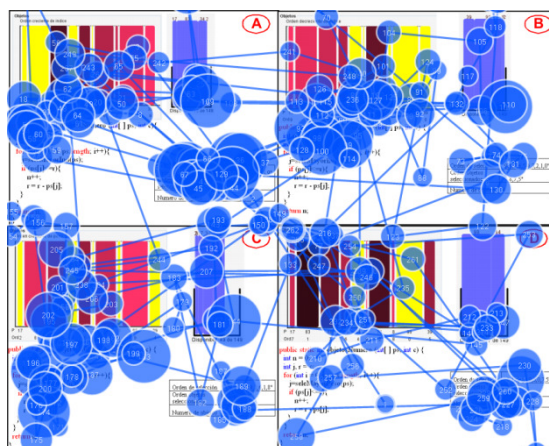


Figura 3 – Ruta de escaneo.

1.2. Objetivos del Trabajo

La meta consiste en el desarrollo de una aplicación para iPad mediante la cual se puedan acceder a las representaciones gráficas de los resultados de los proyectos de *eye tracking*.

Así pues, el objetivo principal consiste en el análisis y tratamiento de los datos que genera un *eye tracker* en una sesión de seguimiento ocular, y su posterior representación gráfica en un iPad, donde se pretende mejorar la experiencia del usuario aprovechando la interacción táctil, la usabilidad y la comodidad y portabilidad que ofrecen este tipo de dispositivos.

La elección del iPad como dispositivo sobre el que desarrollar la aplicación se basa en el constante crecimiento que hoy en día están teniendo las aplicaciones para dispositivos móviles. La comodidad de uso, la facilidad de interacción y las grandes posibilidades de este tipo de dispositivos son las grandes bazas que están haciendo de éste un mercado en alza y unas tecnologías muy atractivas sobre las que trabajar, estudiar o, simplemente, pasar un rato de ocio.

Por otra parte, no solo se ha elegido el iPad por lo que ya constituye en sí, sino también debido a que el software que se encarga de representar los datos resultantes del *eye tracker* (Tobii Studio [Tobii Technology AB Diciembre de 2014]), únicamente se encuentra disponible para ordenadores bajo el sistema operativo Microsoft Windows; además, es un software con licencia de pago. Por ello, lo que se pretende es extender las posibilidades de representación y estudio de estos proyectos de *eye tracking* a través de otros tipos de dispositivos, como los dispositivos móviles (en este caso, el iPad), y otros sistemas operativos (iOS en cuestión).

1.3. Marco de Aplicación

La aplicación desarrollada está orientada a su uso por parte de grupos de investigación en los que su labor o parte de ella se base en el estudio de la usabilidad mediante técnicas de *eye tracking*; como es el caso del grupo de investigación CHICO de la Escuela Superior de Informática de Ciudad Real de la Universidad de Castilla-La Mancha.

Así pues, se espera que esta aplicación tenga cierta repercusión en este ámbito y que su uso lleve a la mejora de los estudios e investigaciones relativas a la usabilidad; así como a la aparición de nuevas aplicaciones relacionadas con este campo y mejoras/actualizaciones en la aplicación en cuestión.

2. Antecedentes, Estado del Arte

2.1. Tobii Studio

Tobii Studio [Tobii Technology AB Diciembre de 2014] es el software necesario para la creación de proyectos de *eye tracking* para el *eye tracker* fabricado por la misma empresa, y, por supuesto, la realización de estos proyectos con usuarios. Cada proyecto puede contar con uno o varios test; y, a su vez, cada test puede contar con diferentes elementos multimedia que se mostrarán al usuario y sobre los que se quiere llevar a cabo el estudio de seguimiento ocular.

Sin embargo, éste no es el objeto de estudio actual, sino la parte de este software que se encarga de representar en un ordenador (bajo el sistema operativo Microsoft Windows) los resultados de un determinado proyecto de *eye tracking*.

Actualmente, la última versión de este software y sobre la que se va a basar todo el desarrollo es la 3.3. Por otra parte, es importante destacar que se trata de un software privativo, por lo que su uso queda supeditado al pago de una licencia.

Por último, cabe destacar que Tobii Studio genera un archivo “.tsv”¹ por cada sesión de *eye tracking* que se realiza. Este archivo cuenta con una multitud de datos recopilados durante dicha sesión de *eye tracking*. En cada conjunto (cada fila) suele haber más de 80 datos diferentes, y una sesión normal de *eye tracking* puede generar más de 100.000 conjuntos de datos.

2.2. Tecnologías de Desarrollo

En primer lugar, el dispositivo sobre el que se ha llevado a cabo el desarrollo de la aplicación en cuestión, es el iPad. El tamaño mayor del dispositivo, que influye en una pantalla más grande, ha sido una de las motivaciones para elegir este dispositivo móvil, ya que las representaciones que se van a hacer se van a ver mucho mejor y va a ser mucho más práctico que habiendo orientado el desarrollo, por ejemplo, para iPhone.

En segundo lugar, iOS es el sistema operativo que utilizan los dispositivos móviles de la marca Apple, como son iPhone, iPad y iPod Touch; y, es por ello, por lo que es el sistema sobre el que se ha desarrollado la aplicación. En concreto, se ha trabajado sobre iOS 8.

En tercer lugar, para poder desarrollar aplicaciones para iOS, al igual que para cualquier otro sistema, es necesario contar con un entorno de desarrollo integrado (IDE, por sus siglas en inglés); en este caso se trata de Xcode. Éste es el entorno de desarrollo integrado de Apple, específico para el desarrollo de aplicaciones para iOS [Apple Inc. Septiembre de 2014] y OS X.

Por último, se ha elegido Objective-C como lenguaje de programación. Éste es el lenguaje de programación secundario en Xcode (tras la aparición de Swift a mediados del año 2014) para el desarrollo de aplicaciones para iOS y OS X.

3. Resultados

3.1. Problemática Inicial

En primer lugar, cabe destacar que, durante las fases iniciales del proyecto, se llevó a cabo un análisis de riesgos en el que se vio como un riesgo potencial los límites de memoria de 1 gigabyte del iPad. Esto podría suponer un inconveniente a la hora de leer, gestionar y guardar los datos relativos a una sesión de seguimiento ocular, ya que los archivos “.tsv” generados por Tobii Studio contienen una cantidad ingente de estos datos.

¹ *Tab-Separated Values*: tipo de archivo con formato de texto simple que almacena datos en una estructura tabular, en el que cada valor se separa del siguiente por un carácter de tabulación.

Debido a este riesgo, se desarrolló una aplicación de prueba para iPad, encargada de leer los datos del archivo “.tsv”, gestionarlos y guardarlos para poder acceder posteriormente a ellos de una manera óptima y ordenada. Como se había predicho, al ejecutar la aplicación y ver los consumos de memoria, se superó el límite de 1 gigabyte de memoria con el que cuenta el iPad.

Por ello, se tomó la decisión de crear dos aplicaciones para poder cubrir todos los objetivos (una para OS X y otra para iOS).

Por una parte, la aplicación para OS X (llamada *iTrackEye Preparer*) se encarga de leer y procesar los datos generados por el *eye tracker*, para generar una estructura de directorios y archivos con estos datos, que sea fácilmente manejable por la aplicación para iPad.

Por otra parte, la aplicación para iOS (con el nombre original, *iTrackEye*) se encarga de procesar estos datos, ahora organizados para un trabajo más óptimo y sin problemas de memoria, y representar los resultados oportunos.

A continuación se muestran varias vistas de ejemplo de las aplicaciones *iTrackEye Preparer* e *iTrackEye*, a través de las cuales se puede observar cómo están desarrolladas, su interfaz gráfica, así como las representaciones que realiza ésta última.

3.2. Aplicación *iTrackEye Preparer*

Al ejecutar *iTrackEye Preparer* se muestra la pantalla inicial del programa, desde donde, aparte de ejecutar la funcionalidad de generación de los datos (mediante el botón “Generate”), se puede seleccionar el archivo “.tsv”, el directorio donde se encuentran los elementos multimedia y el directorio de salida donde se creará la carpeta con todos los datos generados.

Por defecto, la aplicación toma como directorio de los elementos multimedia la carpeta donde se encuentra el archivo “.tsv”, y como directorio de salida el escritorio del ordenador. Lógicamente, estas rutas se pueden cambiar a gusto del usuario, dependiendo de dónde estén dichos archivos multimedia o dónde se quiera que se generen los datos.

En la Figura 4 se puede observar la pantalla inicial que muestra la aplicación *iTrackEye Preparer* al ejecutarse.

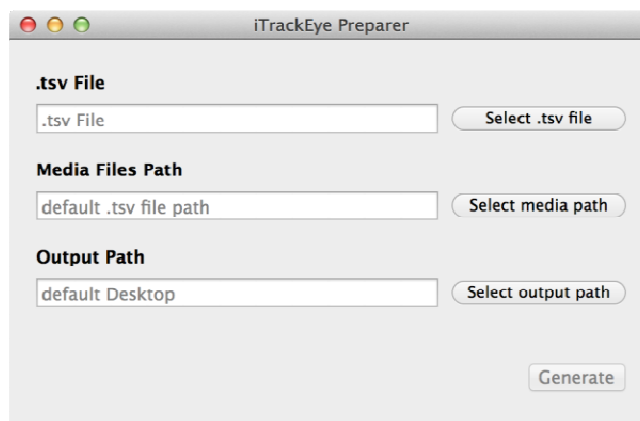


Figura 4 – Pantalla inicial de *iTrackEye Preparer*.

3.3. Aplicación *iTrackEye*

Todos los datos y estructura que genera la aplicación *iTrackEye Preparer* se deben importar a la aplicación de iPad. Para ello se decidió que estos datos se deberían subir a la carpeta de Dropbox del usuario de la aplicación. Por lo que la aplicación de iOS deberá leer dichos datos desde Dropbox. En la Figura 5 se puede observar la vista que aparece al intentar conectar la aplicación *iTrackEye* con la cuenta de Dropbox del usuario.

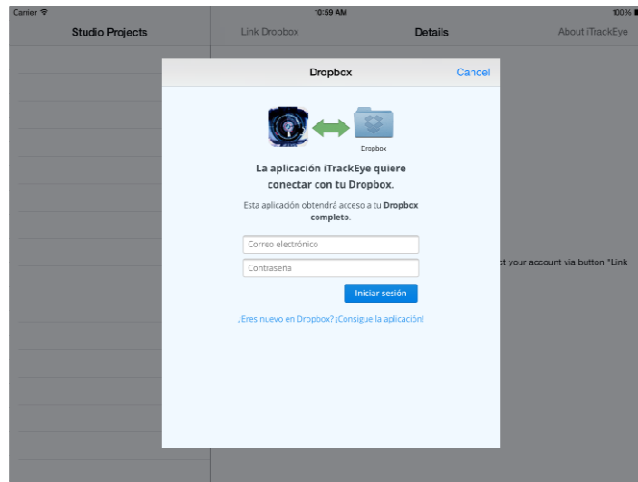


Figura 5 – Enlace de *iTrackEye* con Dropbox.

Ya en la aplicación *iTrackEye*, al iniciarla se descargan los datos necesarios desde Dropbox y se muestra la pantalla inicial de la aplicación, tal y como se muestra en la Figura 6, desde donde se puede empezar a interactuar con la misma.

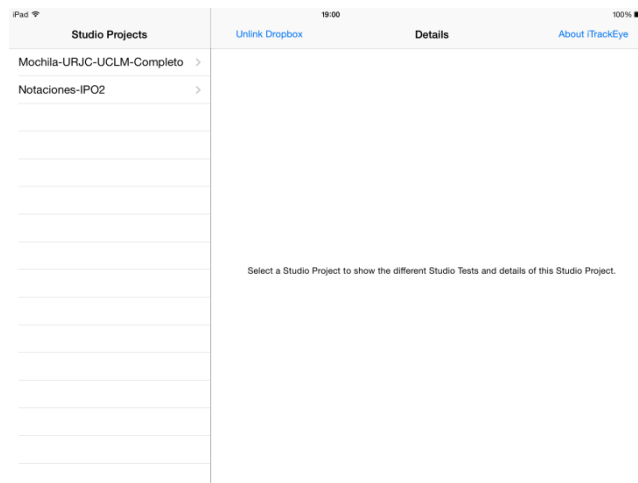


Figura 6 – Pantalla inicial de *iTrackEye*.

Desde esta vista se puede enlazar o desenlazar la cuenta de Dropbox del usuario (mediante los botones “Link Dropbox” y/o “Unlink Dropbox”), acceder a la información y ayuda disponible sobre la aplicación (mediante el botón “About iTrackEye”), o seleccionar un determinado “proyecto” y, dentro de éste, un “test” y un “participante” (a través de la lista de selección existente en la parte izquierda de la pantalla).

Cuando se selecciona un determinado “participante” desde la lista de selección, se muestra la vista que se puede observar en la Figura 7.

En dicha vista se muestran los elementos multimedia del test en cuestión, junto con la información necesaria sobre dicho elemento (nombre, dimensiones y vista preliminar), así como los botones necesarios (“Heat Map”, “Blind Spots Map”, y “Scan Path”) para poder generar el mapa de calor, mapa de puntos ciegos o ruta de escaneo relativo a dicho elemento multimedia y dicho participante.

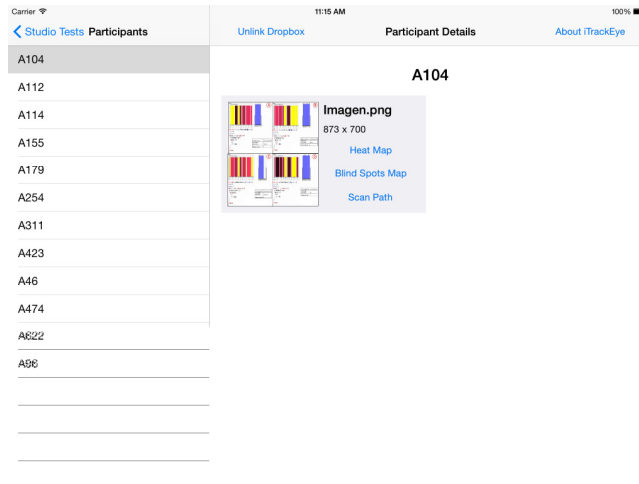


Figura 7 – Vista de detalles de un “participante” en *iTrackEye*.

Cuando se genera un mapa o ruta de un determinado elemento multimedia relativo a un participante, se muestra la vista de resultados con dicho mapa o ruta.

En la Figura 8, la Figura 9 y la Figura 10, se puede ver un ejemplo de la generación del mapa de calor, del mapa de puntos ciegos y de la ruta de escaneo, respectivamente, de un elemento multimedia.

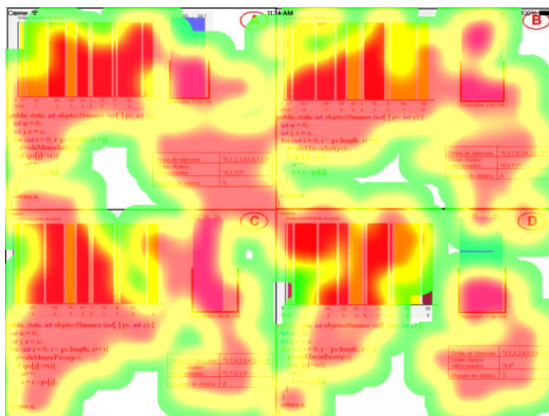


Figura 8 – Mapa de calor generado mediante *iTrackEye*.

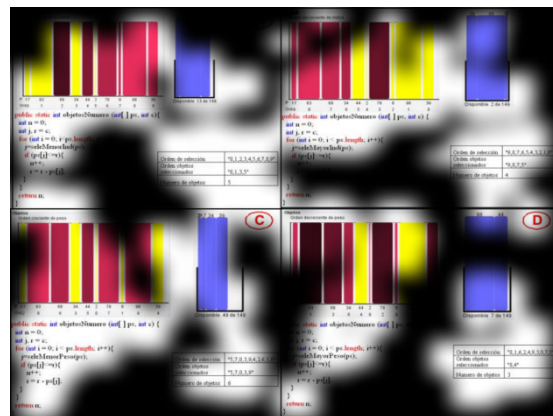


Figura 9 – Mapa de puntos ciegos generado mediante *iTrackEye*.

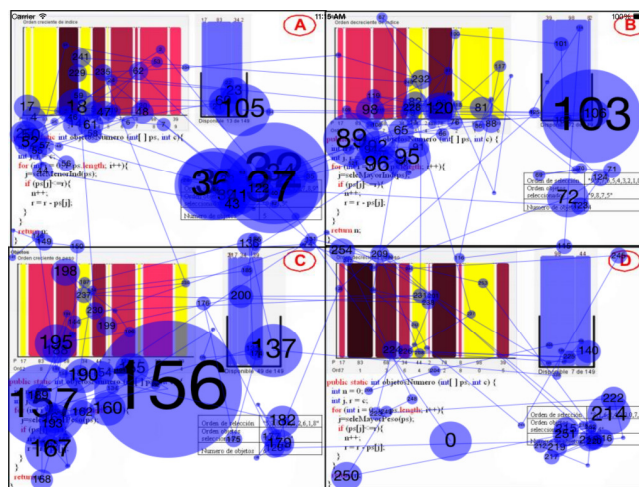


Figura 10 – Ruta de escaneo generada mediante *iTrackEye*.

Si sobre esta vista de resultado se hace un “toque” (se pulsa sobre ella), aparecerán dos barras, tal y como se puede observar en la Figura 11.

Por un lado, la barra inferior contiene información acerca del mapa o ruta generado. Esta información define el proyecto, test, participante y elemento multimedia al que pertenece la generación, así como el tipo de mapa o ruta que se está mostrando.

Por otro lado, la barra superior cuenta con una leyenda (solo en el caso de los mapas, a modo informativo, con los colores utilizados) y con tres botones: uno para exportar el mapa o ruta (“Export”), otro para cerrar la vista (“Done”), y otro para ocultar o mostrar el mapa o ruta (el que tiene forma de ojo).

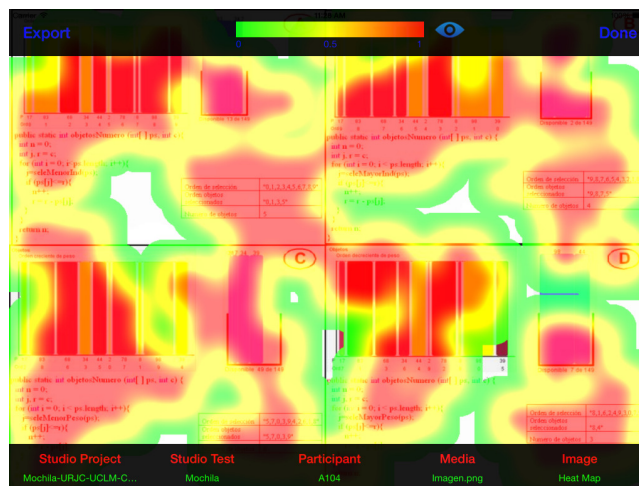


Figura 11 – Vista de resultado con las barras de información y acciones en *iTrackEye*.

3.4. Sobre las Aplicaciones

Es importante destacar que ambas aplicaciones se han desarrollado siguiendo las reglas de oro de Ben Shneiderman [Shneiderman et al. 2010], mediante las cuales establece los ocho principios básicos de usabilidad en el diseño de interfaces.

Individualmente, la aplicación *iTrackEye Preparer* se ha desarrollado siguiendo los principios de diseño de aplicaciones para OS X fijados por Apple [Apple Inc. Octubre de 2014].

Por su parte, para el desarrollo de la aplicación *iTrackEye* también se han seguido los principios de desarrollo y diseño de aplicaciones para iOS establecidos por Apple [Apple Inc. Septiembre de 2014] [Apple Inc. Diciembre de 2014], así como los patrones de usabilidad para dispositivos móviles definidos por Jacob Nielsen [Nielsen et al. 2013].

4. Conclusiones

La aplicación *iTrackEye* [Patón Junio de 2015] ha nacido con un marcado carácter investigador, ya que, gracias a la representación de los resultados de una sesión de *eye tracking* que lleva a cabo, es una herramienta muy útil para aquellos usuarios que trabajen en este campo del análisis o investigación de la usabilidad.

Durante toda la implementación se ha pretendido conseguir que tanto *iTrackEye* como *iTrackEye Preparer* sean aplicaciones intuitivas, rápidas, que permitan al usuario interactuar de la manera más sencilla posible, dándole las funcionalidades que necesita para que su trabajo no se convierta en una tarea ardua.

En definitiva, se han conseguido crear dos aplicaciones simples, elegantes, limpias y atractivas, permitiendo al usuario centrarse más en la funcionalidad que en el aspecto.

4.1. Pruebas de Usabilidad

Ambas aplicaciones se han desarrollado siguiendo los principios de Interacción Persona-Ordenador [Hewett et al. 1992], teniendo en cuenta diferentes reglas y/o guías que atañen a aplicaciones de este tipo.

Por ello, se han llevado a cabo una serie de pruebas de usabilidad [Patón Julio de 2015] con el fin de comprobar la correcta adecuación de las aplicaciones a los principios de Interacción Persona-Ordenador (en especial de la aplicación principal: *iTrackEye*).

Estas pruebas se realizaron con una serie alumnos e investigadores de la Escuela Superior de Informática de Ciudad Real, y consistieron en plantear una serie de actividades a realizar en la aplicación *iTrackEye* con el dispositivo iPad, mientras se registraba y grababa toda la actividad con el *eye tracker*.

Antes y después de realizar las pruebas sobre la aplicación, los participantes debían rellenar un cuestionario acerca de sus conocimientos sobre representaciones mediante la técnica de *eye tracking* y sobre su experiencia con la aplicación.

Tras analizar toda la información, se puede determinar que la interfaz gráfica de usuario, así como las representaciones llevadas a cabo en la aplicación *iTrackEye*, siguen los principios de Interacción Persona-Ordenador; ya que en estas pruebas de usabilidad realizadas, los resultados recopilados de los usuarios dan a entender su satisfacción en relación a este ámbito.

4.2. Líneas de Trabajo Futuro

En primer lugar, la aplicación *iTrackEye* se ha limitado a su implementación únicamente para iPad, pero podría implementarse también para otro tipo de dispositivos móviles, no necesariamente de marca Apple, o incluso también para ordenadores. Cuanto más dispositivos móviles u ordenadores de diferentes marcas y sistemas operativos soporten *iTrackEye*, es decir, cuanto más multiplataforma sea, mucho mejor para la expansión de esta aplicación.

De hecho, actualmente el proyecto se encuentra en una fase de expansión en la que se está desarrollando una versión de *iTrackEye* para web; pudiendo ser accesible desde todo tipo de dispositivos móviles u ordenadores de cualquier índole y con cualquier sistema operativo.

En segundo lugar, *iTrackEye* únicamente funciona mostrando, como resultados de la generación de los mapas y rutas, imágenes estáticas; sin embargo se podrían representar también vídeos donde se vea segundo a segundo cómo se ha ido fijando el participante en los elementos mostrados durante la sesión de *eye tracking*.

En tercer lugar, en las rutas de escaneo surge un problema que radica en la gran cantidad de fijaciones que realiza un usuario al explorar visualmente una escena en concreto; por lo que esta representación es compleja y resulta difícil apreciar con claridad el orden representado.

Así pues, es interesante desarrollar una funcionalidad que permita simplificar las rutas de escaneo, representado únicamente aquellos datos que puedan resultar más relevantes para el investigador o analista, es decir, crear un método de reducción y abstracción de la información acerca del número de fijaciones que se recogen en una ruta de escaneo.

Actualmente, el grupo de investigación CHICO de la Escuela Superior de Informática de Ciudad Real de la UCLM está desarrollando esta nueva funcionalidad en la aplicación *iTrackEye*, donde ya se han conseguido los primeros resultados.

Por último, surge una nueva idea de un estudio realizado en la Universidad de Padova (Italia), relacionado con la identificación de patrones de visualización [Mason et al. 2013].

Esta identificación y/o detección de patrones de visualización debe permitir, por un lado, ver qué patrones de visualización ha seguido un determinado participante (comprendiendo el orden

de exploración visual), y, por otro lado, poder determinar si los participantes han seguido un patrón en concreto en sus fijaciones en un momento determinado o tras una acción en concreto (esto es, identificar similitudes en el comportamiento de exploración visual entre ellos).

Al igual que con la funcionalidad de reducción de rutas de escaneo, el grupo de investigación CHICO está trabajando sobre esta nueva funcionalidad relacionada con la generación y/o identificación de patrones de visualización.

5. Agradecimientos

Esta investigación ha sido parcialmente financiada por el Ministerio de Ciencia e Innovación a través del proyecto TIN2011-29542-C02-02, la Red Temática CYTED 513RT0481 y los Proyectos de la JCCM PPEII-2014-012A y PPII11-0013-1219.

6. Referencias

- Hewett, T. T., Baecker, R., Card, S., Carey, T., Gasen, J., Hefley, B., Mantei, M., Perlman, G., Strong, G., and Verplank, W., *ACM SIGCHI Curricula for Human Computer Interaction*, The Association for Computer Machinery, Broadway, NY, USA, 1992.
- Mason, L., Tornatora, M. C., and Pluchino, P., “Do fourth graders integrate text and picture in processing and learning from an illustrated science text? Evidence from eye-movement patterns”, In *Journal of Computers & Education*, nº 60, 2013, 95-109.
- Nielsen, J. and Raluca, B., *Mobile Usability*, New Riders, Berkeley, CA, USA, 2013.
- Shneiderman, B. and Plaisant, C., *Designing the User Interface: Strategies for Effective Human-Computer Interaction (5th Edition)*, Addison-Wesley Publishing Co., Reading, MA, USA, 2010.
- Apple Inc., *iOS App Programming Guide*, <https://developer.apple.com/library/ios/documentation/iPhone/Conceptual/iPhoneOSProgrammingGuide/Introduction/Introduction.html>, (09 de Junio de 2015), Septiembre de 2014.
- Apple Inc., *iOS Human Interface Guidelines*, <https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/index.html>, (09 de Junio de 2015), Diciembre de 2014.
- Apple Inc., *OS X Human Interface Guidelines*, <https://developer.apple.com/library/mac/documentation/UserExperience/Conceptual/OSXHIGuidelines/index.html>, (09 de Junio de 2015), Octubre de 2014.
- Patón, J. D., *Wiki of iTrackEye*, http://chico.inf-cr.uclm.es/itrackeye/wiki/index.php/Página_principal, (09 de Junio de 2015), Junio de 2015.
- Patón, J. D., *Usability Tests of iTrackEye*, http://chico.inf-cr.uclm.es/itrackeye/wiki/index.php/Pruebas_de_usabilidad, (08 de Julio de 2015), Julio de 2015.
- Tobii Technology AB, *User Manual - Tobii Studio*. Manual Version 3.3.0, http://www.tobii.com/Global/Analysis/Downloads/User_Manuals_and_Guides/Tobii%20Studio%203.3%20User%20Manual.pdf, (09 de Junio de 2015), Diciembre de 2014.