

ALGORITMOS PARA BÚSQUEDA Y UBICACIÓN DE PATRIMONIO CULTURAL CON REALIDAD AUMENTADA USANDO DISPOSITIVOS MÓVILES

Javier Ignacio Tomaselli ¹

Universidad Columbia del Paraguay - Paraguay

Heriberto Pintos Correa²

Universidad Columbia del Paraguay – Paraguay

<http://orcid.org/0000-0002-5966-2266>

Recibido: 28/09/2019

Aprobado: 30/10/2019

Resumen

Se presenta el diseño y la implementación de un sistema de realidad aumentada que permite conocer un patrimonio cultural a través de realidad virtual, con el objeto de brindar la posibilidad de satisfacer la necesidad de información acerca de la historia nacional. La aplicación busca difundir y proteger la diversidad cultural estimulando el interés de la ciudadanía y de los viajeros para que amplíen sus conocimientos históricos. Empleando una metodología cualitativa y descriptiva, se aplican algoritmos para la búsqueda de ubicación del patrimonio que, con el concepto de realidad aumentada, brinda información al usuario. Para ello se establecen la orientación del dispositivo y las distancias entre diferentes puntos en un sistema de coordenadas geográficas. Está diseñada para el uso en dispositivos móviles, en las pruebas de funcionamiento se incluyen algunos lugares históricos en la base de datos junto con aspectos que hacen al patrimonio cultural. Incorpora con el uso de sensores una innovadora experiencia de

¹ Ingeniero en Informática – Universidad Columbia del Paraguay. Líder técnico de Infocenter S.A. Correo: javier.tomaselli@gmail.com

² Master en Ciencia de la Computación - Universidade Federal do Rio Grande do Sul. Lic. en Análisis de Sistemas – Facultad Politécnica, Universidad Nacional de Asunción. Docente de la carrera de Ingeniería en Informática de la Universidad Columbia del Paraguay. Correo: heriberto.pintos@columbia.edu.py

usuario que permite acceder al contenido de forma simple y atractiva. La utilización de esta nueva tecnología, que combina en un escenario real contenido virtual, está al alcance de los usuarios de teléfono inteligente o tableta que funcionen con el sistema android. Esta investigación forma parte del trabajo de conclusión de carrera de la Universidad Columbia del Paraguay.

Palabras clave: Sistemas de información. Realidad aumentada. Patrimonio cultural en dispositivo móvil. Sistema de posicionamiento. Geolocalización cercana.

Abstracts

The design and implementation of an augmented reality system that allows knowing a cultural heritage through virtual reality is presented, in order to provide the possibility of satisfying the need for information about national history. The application seeks to disseminate and protect cultural diversity by stimulating the interest of citizens and travelers to expand their historical knowledge. Using a qualitative and descriptive methodology, algorithms are applied to search for the location of the heritage that, with the concept of augmented reality, provides information to the user. To do this, the orientation of the device and the distances between different points in a geographic coordinate system are established. It is designed for use on mobile devices, in the functional tests some historical places are included in the database along with aspects related to cultural heritage. With the use of sensors, it incorporates an innovative user experience that allows access to content in a simple and attractive way. The use of this new technology, which combines virtual content in a real setting, is available to smartphone or tablet users that run on the android system. This research is part of the completion work of the Columbia University of Paraguay.

Keywords: Information systems. Augmented reality. Cultural heritage on a mobile device. Positioning system. Nearby geolocation.

Apañuâi jekuaapyre jaheka ha ñamohenda hagua mba'eteekuéra culturapegua ko añetegua ñembotuicháva jaiporuvo pumbyrymikuéra Ñemombyky

Ojehechauka ko ta'anga ha petei hetamba'e kuera jaipuruhagua ko aÑetega ñembotuicháva, ohejava kuaa ko mba'e teekuera culturapegua pe aÑetega ikatuvaêragua'u rupive, ohupytyhagua ha ome'ehagua pe mba'e ikatuvo oiko ikatuhaguaicha omongapyhy pe tekoteve momaranduregua tembiasaka tetã rehegua. Ko moiha oheka omosarambi ha oñangareko hagua pe opaichagua arandukuaa ombokyre'yvo pe pyapy tetagua ha umi guataharakuérare ikatuhaguaicha omomtuicha pe jekuaa tembiasakue rehegua. Oiporuvo petei jejapo hagua hyepypegua ñemoñe'ê oje'éva, oñemoi apañuái jekuaapyre ñambohenda hagua mba'eteekuera pe, he'iséva aÑetega ñembotuichávare ome'ë hagua momaranduregua umi oiporuva ko hetamba'e kuera. Upevare oñemoi mbohape pe pumbyrymikuera ha pe mombryrykuera ha umi ojavýva ha'etepekuera pe hetamba'e kuera nderaháva vyapekuaape. Ojejapo oijeporuhagua pumbyrymikuérape, pe ñeha'ähápe oikopa oñemoingue oimerae tenda tembiasakue pe pumbyrymikuera pe umi ambue ojavova mba'eteekuera culturapegua. Omoingue avei pe oijeporuhaguame pe pumbyry oñanduba petei ñembopyahu tembiasakue arandukuera mba'éva ome'eva ñandeve jaike hagua umi oguerékóva ypye ysaja ka'avo. Ko jeiporu mbaépuahu mbae rehegua, ombojehe'áva petei tenda aÑetega oguereko ha ikatuvaêragua'u, oï enterovéva oiporuva pumbyrymikuera po guype oguerékóva hetamba'e kuera android. Ko jetypeka ha'e pe tembiapo paha rehegua ko carrera Universidad Columbia Paraguay regua.

Ñe'ë tekotevéva: Hetamba'e kuera momaranduregua. AÑetega ñembotuicháva.

Mba'eteekuera culturapegua. Hetamba'e kuera ñeihame. Yvyjehuha aguigua.

Introducción

El estado del arte en el internet de las cosas y la masificación del uso de los teléfonos inteligentes permiten incorporar paradigmas emergentes que se abren paso y continúan creciendo en la actualidad. La combinación de dos conceptos, realidad aumentada y geolocalización, incorporados en tecnología móvil pueden ser aplicados en una diversidad de soluciones, como ser el de acercar a las personas de inmediato a un patrimonio cultural, pues “el patrimonio cultural de un pueblo comprende las obras de sus artistas.” UNESCO (1982, p.43).

Este trabajo muestra el diseño y la implementación de un sistema de realidad aumentada que permite conocer un patrimonio cultural a través de las plataformas web y móvil, con el objeto de brindar la posibilidad de satisfacer la necesidad de información acerca de la historia nacional. La aplicación busca difundir y proteger la diversidad cultural estimulando el interés de la ciudadanía y de los viajeros para que amplíen sus conocimientos históricos.

Se emplea una metodología cualitativa y descriptiva, para la búsqueda de ubicación del patrimonio y se aplican algoritmos basados en modelos matemáticos que con el concepto de realidad aumentada brinda información al usuario.

Las tecnologías de realidad aumentada y geolocalización integradas en una herramienta digital posibilita la búsqueda de ubicaciones patrimoniales de Paraguay a través de dispositivos móviles, da acceso a los usuarios a una diversidad de datos del patrimonio cultural, además de permitir la interacción con esos bienes en tiempo real, recreando una experiencia innovadora y, por consecuencia, se logra la transmisión de valores de dicho patrimonio.

La solución propuesta está orientada a residentes y turistas para mejorar su experiencia en informarse sobre un determinado patrimonio cultural de nuestro país desde su dispositivo móvil, ya que “el patrimonio cultural del Paraguay se encuentra constituido por los bienes muebles e inmuebles, materiales e inmateriales, ambientales y construidos, seculares o eclesiásticos, públicos o privados, en cuanto resulten relevantes para la cultura” (Ley N° 5621, 2016, p.1).

El ranking elaborado por el Observatorio Turístico Nacional de Paraguay señala un incremento de turistas de más del 50% entre los años 2010 y 2015 (Observatorio Turístico Nacional, s.f.). En este documento se describen los conceptos de las tecnologías asociadas y cálculos matemáticos, se presenta la arquitectura de la solución y el detalle de los procesos que hacen a la funcionalidad de la solución.

La aplicación es alimentada con datos de patrimonios culturales e históricos precargados que forman parte de las pruebas de funcionamiento en ambiente web y móvil.

Realidad Aumentada, realidad virtual y realidad mixta

La Realidad Aumentada (RA) es una variación de Realidad Virtual (RV). Las tecnologías de RV posicionan a la persona dentro de ese entorno, a diferencia de la RA que posibilita ver el mundo real, donde se superponen o componen objetos virtuales (Morcillo, Vallejo, Albusac y Castro, 2013. Prefacio). De acuerdo con Milgram y Kishino (1994) la Realidad mixta abarca el espacio entre los extremos de lo real y virtual.

Ronald Azuma (1997), define RA como sistemas que deben tener tres características principales: combina elementos reales y elementos del mundo virtual, es interactiva en tiempo real, y la información es almacenada en 3D.

Lens-Fitzgerald (2009) clasifica la RA en cuatro niveles como una forma de medir la complejidad de las tecnologías involucradas. A más nivel, mayores son las posibilidades de las aplicaciones (Prendes, 2015).

Este estudio de caso se enmarca en RA de nivel 2 sin marcadores, que funciona sin necesidad de un marcador asociado como requisito. Los elementos utilizados para este nivel son el Sistema de Posicionamiento Global (GPS) y la brújula. La información es proveída por el dispositivo móvil a través de sensores y las coordenadas geográficas se superpondrán en pantalla como Puntos De Interés (PDI) en imágenes del mundo real, teniendo en cuenta las limitaciones de alcance del GPS que menciona Kevin Bonsor (2011).

Teoría de la Geodesia

Esta teoría permite detectar PDI siguiendo una serie de pasos. La aplicación se constituye en base a la geodesia en la cual se prescinde de la RA que reconoce marcadores.

El azimut es el ángulo medido en sentido horario en un plano y línea de referencias desde el observador hasta un PDI proyectado en ese plano, es el ángulo entre la dirección actual de la brújula y el norte magnético. En otras palabras, si el borde del dispositivo se orienta hacia el norte magnético, el azimut es 0 grados.

Para detectar un PDI precargado y visualizarlo en pantalla, hay que ubicar ese punto comparando el ángulo azimut real con el ángulo azimut teórico calculado previamente. Es necesario para determinar un PDI obtener la ubicación del GPS del

dispositivo, calcular el azimut real del dispositivo a través de los datos recibidos de sus sensores y obtener la ubicación del GPS del PDI a buscar, que está precargado con su respectiva ubicación. Posteriormente se debe calcular el azimut teórico basado en los datos del GPS obteniendo las diferencias entre los puntos de origen y destino y comparar ambos resultados.

Puede existir una variable de precisión que establece un límite inferior y un límite superior para ubicar el resultado. Para realizar estos cálculos de diferencias se ignora la curvatura de la tierra y los puntos se tratan en un plano. El valor del azimut se calcula obteniendo el ángulo entre P1 y P2, donde P1 es el dispositivo y P2 el PDI a buscar (Fig. 1). El ángulo azimut a calcular se observa como el ángulo α de un triángulo rectángulo.

Las coordenadas geográficas están determinadas en la tierra por su latitud y longitud. El dispositivo es el origen P1 y el PDI a buscar es el destino P2. La diferencia de latitud determina el arco de meridiano entre los paralelos del P1 origen y PDI destino.

$$\Delta lat = lat_2 - lat_1$$

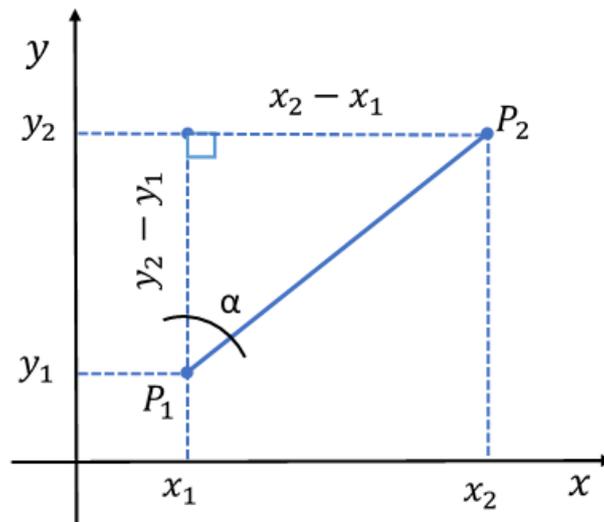
$$\Delta lat = \Delta y = y_2 - y_1$$

Mientras que la diferencia de longitud determina el arco entre origen y destino.

$$\Delta long = long_2 - long_1$$

$$\Delta long = \Delta x = x_2 - x_1$$

Figura 1. *Ángulo Azimut en el plano.*



Luego de obtener las diferencias, se busca la tangente de la división y la resultante del cálculo corresponde al azimut teórico.

$$\tan \alpha = \left| \frac{\Delta x}{\Delta y} \right|$$

Finalmente se calcula el valor del azimut real, que se obtiene con los sensores del dispositivo. Se comparan ambos valores de acuerdo con un límite de tolerancia fijo y si la diferencia entre ambos está dentro del límite, entonces el PDI se visualiza en pantalla.

Sensores

Los sensores permiten recopilar información del mundo exterior sin procesar, pero con alta precisión y con la velocidad que es determinada por el desarrollador que se manipula de forma homogénea. La forma de acceder es a través de la clase `SensorManager`, cuando se instancia un objeto, permite diferenciar el tipo de sensor y establece el retraso de recuperación de dichos valores por tipo de sensor. Una vez instanciado, el objeto de esta clase implementa `SensorEventListener` del tipo *listener*, que permite obtener los datos del sensor cuando se producen cambios en diferentes eventos a través de la implementación de los métodos obligatorios.

En la aplicación son utilizados sensores de movimiento y de posición basados en hardware que son componentes físicos electrónicos integrados a los dispositivos. El

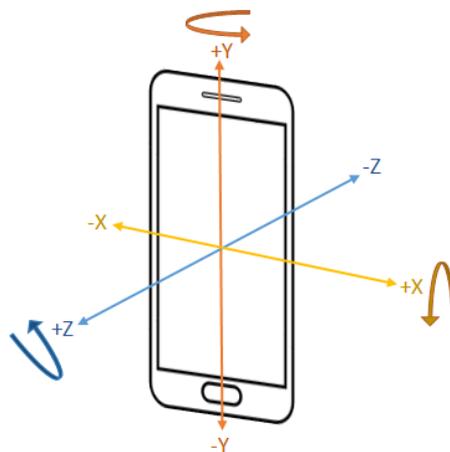
entorno permite solamente la utilización de los datos de sensores y la ubicación de PDI en pantalla en una orientación vertical. Para realizar esta activación al inicio de la actividad principal en el método onCreate se establece dicha propiedad. Se utiliza el sensor de movimiento que mide la fuerza de aceleración en los diferentes ejes, incluida la fuerza de gravedad, permite detectar movimientos en el dispositivo y obtiene su orientación combinándolo con el magnetómetro. La frecuencia de actualización de este sensor va a ser normal y por cada cambio que exista se obtendrán los nuevos valores constantemente.

Magnetómetro

Este sensor tipo brújula, mide el campo geomagnético en los tres ejes físicos (x, y, z). La frecuencia de actualización de este sensor va a ser del tipo juego. Esta frecuencia es un poco más rápida para la obtención de valores.

Se pueden combinar los datos del magnetómetro con el acelerómetro y así obtener una orientación más precisa (Fig. 2). Para suavizar los movimientos obtenidos y que los cambios no sean tan bruscos se declara una variable y se multiplica cada vez que se registran valores.

Figura 2. Ejes magnetómetro.



Vector de rotación

Este sensor mide la orientación de un dispositivo y a través de sus datos recuperados se obtiene el ángulo azimut. Cada sensor tiene una tasa de actualización adecuada a la interfaz del usuario.

La frecuencia de los sensores es registrada en el evento onStart y esta permite registrar los cambios producidos en el evento onSensorChanged. Con los datos obtenidos del acelerómetro y del magnetómetro se forma una matriz de rotación, que traduce los datos del sensor de un sistema de coordenadas a otro. En este caso, la traducción es del sistema de coordenadas del dispositivo al sistema de coordenadas de la tierra. Cada uno de los puntos de los tres ejes se representa como un vector 3D. Con esta matriz, se calcula el ángulo de giro en cada registro de cambio que exista en los sensores y con el sensor del vector de rotación se obtiene el ángulo azimut. Es necesario convertir los valores del vector a una matriz y de esta forma se vuelven a reasignar los valores a otro sistema de coordenadas para que la orientación natural del dispositivo no influya.

La precisión de los sensores desafortunadamente no es perfecta, principalmente debido al campo magnético emitido por el propio dispositivo (Pshics, 2019).

Algoritmo de Haversine

Para calcular la distancia entre dos puntos en un plano se aplica el teorema de Pitágoras. Esta tarea resulta más compleja cuando la distancia está dada por dos puntos sobre la superficie terrestre. Los datos en este caso son, las coordenadas geográficas, latitud y longitud. Se utiliza el método de Haversine o semiverseno (Movable Type Scripts, 2019).

La fórmula de Haversine es una de las ecuaciones más importantes para la navegación astronómica y su aplicación es la siguiente:

$$d = 2r \arcsin \left(\sqrt{\sin^2 \left(\frac{\phi_2 - \phi_1}{2} \right) + \cos(\phi_1) \cos(\phi_2) \sin^2 \left(\frac{\lambda_2 - \lambda_1}{2} \right)} \right)$$

Donde d es la distancia entre dos puntos, r es el radio de la tierra, ϕ_1 es la latitud del punto de origen, ϕ_2 es la latitud del punto de destino, y λ es la diferencia de longitudes. El radio medio es 6.371 km. Como la tierra no es una esfera perfecta, el radio en los polos varía. Por lo tanto, esta función de distancia es sólo una aproximación.

Entorno de realidad mixta y patrimonio cultural

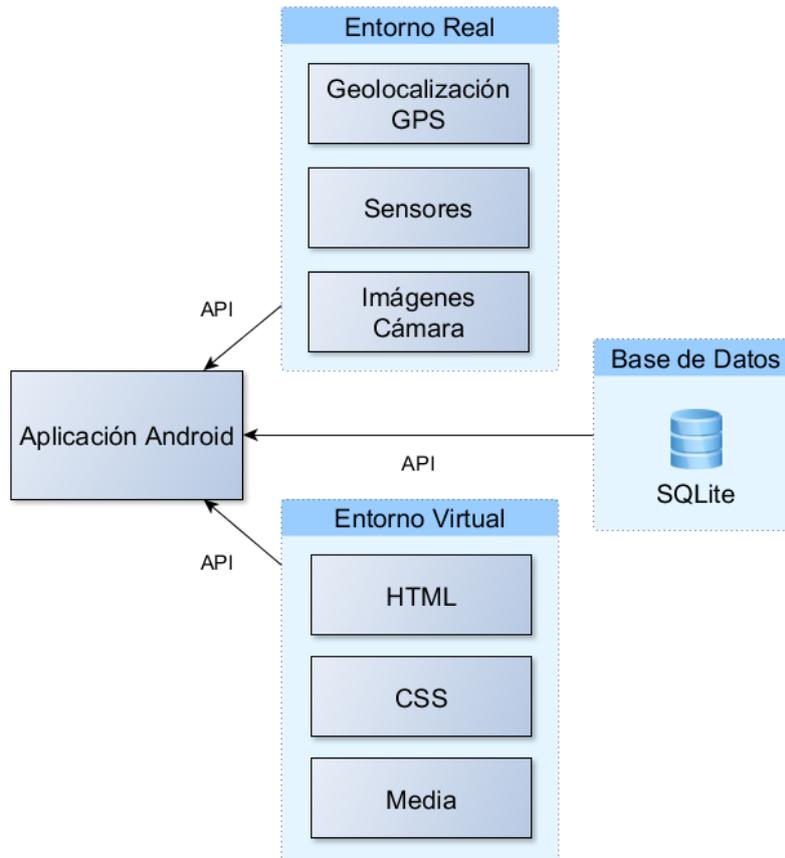
La aplicación soporta el sistema operativo móvil Android (Fig. 3) versiones 7.0 en adelante (Android nougat) compiladas con la Application Programming Interface (API) (Android y Google, 2017).

La arquitectura es cliente-servidor y en la interfaz gráfica de usuario se visualizan el entorno real y el entorno virtual contemplados en un sistema de RA de nivel 2, utilizando como activador del entorno virtual el sensor del GPS del dispositivo. Este sensor del GPS y de la brújula sustituyen los marcadores que determinan la localización y orientación del usuario en tiempo real.

Se utiliza Android studio 3.2, entorno de desarrollo integrado oficial para aplicaciones Android y las herramientas de Android software development kit, facilitando las APIs disponibilizadas por google.

De acuerdo con la agencia de información paraguaya, el 52% del patrimonio histórico se encuentra en el centro de Asunción (AIP, 2017). Se adoptan cinco de ellos como datos de prueba que son visualizados como PDI, que son hipotéticos y referenciados convenientemente; en esta versión es posible visualizar solamente un PDI a la vez.

Figura 3. *Arquitectura de la aplicación móvil.*



Especificaciones funcionales

La aplicación tiene distintas funcionalidades, como ser:

- Visualizar diferentes PDI geolocalizados de acuerdo con la ubicación actual del usuario.
- Desplegar de forma interactiva contenido de algún PDI seleccionado.
- Realizar búsquedas por categorías
- Búsquedas por ciudad y departamento y por año.
- Visualizar por categorías y seleccionar el PDI deseado por el usuario.
- Establecer una distancia máxima para visualizaciones en el dispositivo.

- Parametrizar la posibilidad de recepción de notificaciones de PDI cercanos estableciendo el tiempo entre cada una.

La versión de Android 7.0 o posterior es SQLite permite la comunicación, manipulación de datos y gestión del BD local con funciones y herramientas totalmente incorporadas y nativas permitiendo un acceso más eficiente y consistente de la información.

El motor que utiliza SQLite es el más popular hoy en día en aplicaciones móviles por su pequeño tamaño, simplicidad en la gestión del contenido y por ser de código abierto. Diseñado en lenguaje C, el BD está contenido en un solo archivo, haciendo que el acceso sea más rápido y eficiente en la comunicación. La versión utilizada del BD SQLite es la 3.0, permite la incorporación del tipo binary large objects para alojar el contenido cultural.

La construcción de las interfaces web de usuario está basada en JavaServer Faces (JSF) en su versión 2.3. Este framework contiene componentes que simplifican la comunicación y el binding de los datos entre la página HyperText Markup Language (HTML) y el código Java de la clase en uso. El diseño de la aplicación está basado en la versión 6.2.2 de PrimeFaces, la librería se utiliza para JSF y es de código abierto. Este componente mejora la experiencia del usuario. Las anotaciones de JSF de las clases utilizadas en el diseño del sistema es @ViewScoped, permite almacenar los objetos solamente mientras se mantenga en esa vista sin recargar la memoria permitiendo la escalabilidad de la sesión.

Funcionalidad de la aplicación Java Web

La clase LoginBean representa el ingreso al sistema web. El método loginIngresar realiza la consulta al BD culturadb para validar usuario y contraseña.

Las consultas al BD SQLite se realizan con la conexión al DataSource de la consola payara server. La clase DepartamentosBean se utiliza para la manipulación de departamentos, CategoríasBean manipula categorías, CiudadesBean manipula ciudades y PDIBean administra los PDI hipotéticos que se visualizarán en la aplicación.

La anotación @PostConstruct utiliza el método init, encargado de conectar y realizar el acceso al BD para obtener los datos. Este método es compartido en todas las demás clases y en cada una obtiene los datos de la entidad que necesite para ser visualizados en el momento de la construcción inicial de la página web.

El método editarDatosFila es llamado de la vista HTML cuando se ejecuta el evento rowEdit, devuelve un objeto tipo RowEditEvent conteniendo los datos de la línea modificada. Los datos recibidos son separados en una variable del tipo clase entidad de acuerdo a la vista actual y con esa información se ingresa al BD para modificar los valores mediante la sentencia UPDATE de Structured Query Language (SQL). Reload realiza una actualización de la página web cuando los datos no se han podido actualizar para que la visualización de pantalla sea consistente. Guardar es el método que se ejecuta para realizar un INSERT de SQL para la actualización de datos nuevos en el BD. Este método valida que los datos no existan en el BD para poder grabarlos. EliminarFila recibe en todas las clases un id y un nombre o descripción. Este id es la clave primaria para realizar el DELETE de SQL al BD. La clase debe contener sus propios getters y setters. Cada PDI se crea mediante la selección de su ubicación teórica, la latitud y longitud se seleccionan en un mapa insertado en la vista de la aplicación Java Web con la API de geolocalización de google maps.

API Google Maps

El mapa se habilita mediante una API obtenida en Google Maps platform. Es activada Maps JavaScript API y posteriormente se inserta en el encabezado del código del HTML de la siguiente manera:

```
<script src=http://maps.google.com/maps/api/js?key=XXX
    type="text/javascript">
</script>
```

Incluyendo en el valor key, la clave proporcionada por Google Maps se registra y se crean las credenciales.

La clase que maneja el patrimonio es PatrimoniosBean y se utiliza para la manipulación de patrimonios; una vez generado, se puede modificar mediante la clase PatrimoniosArchivosBean. La última modificación será el contenido principal de visualización del patrimonio dentro del layout principal en la aplicación Android, al ser seleccionado un PDI. Cada patrimonio nuevo insertado en el BD crea a su vez una carpeta que actúa como directorio principal en el documento multimedia, contiene el id del patrimonio creado y en ella se guarda el documento subido con su respectivo nombre y tamaño.

Se incluyen reportes controlados por la clase ReportesBean, en esta opción se grafican las visitas de usuarios divididas en patrimonios y PDI, con gráficos del tipo circular para mostrar porcentajes de visitas. Los reportes son visualizados por el administrador de la aplicación web y no permite a los usuarios generar contenido propio.

Funcionalidad y diseño de la aplicación android

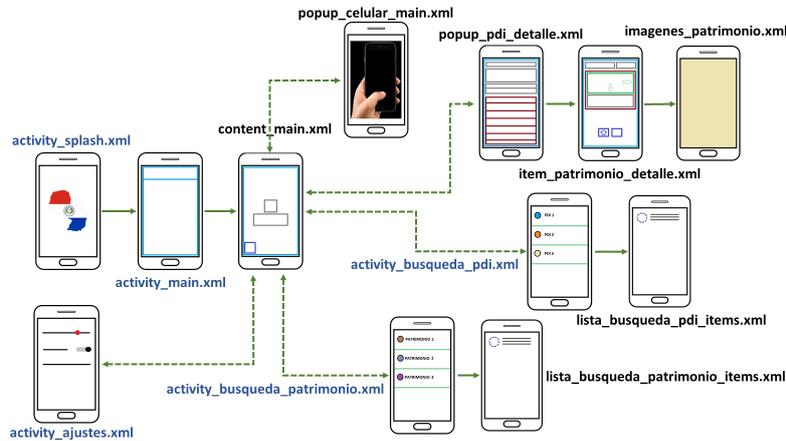
La aplicación tiene una actividad principal donde se desarrollan y ejecutan sus funcionalidades más importantes, interviene el uso de los sensores, la aplicación de la teoría de la geodesia, el uso de la fórmula de Haversine, el layout de los principales componentes y la obtención de los ajustes establecidos por el usuario.

La relación que existe entre todas las actividades y sus respectivos layouts está definida de acuerdo con el mapa layouts (Fig. 4). Cada uno de los layout contiene vistas propias de la aplicación. Existen cinco actividades, la que es automática se ejecuta una sola vez en el inicio y las otras actividades secundarias pueden accederse de forma manual. Las actividades secundarias parten siempre desde la actividad principal y solamente pueden volver a esta.

Cada inicio de la aplicación consta de cinco partes: el cambio por defecto de la orientación del dispositivo a vertical, el acceso al BD para recuperar los datos de entidades, el inicio de los sensores para empezar a leer datos de estos, el establecimiento de la configuración definida por los ajustes de usuario y la ejecución del hilo principal a través de la implementación de la clase SurfaceHolder.Callback.

Cuando el método `surfaceCreated` es llamado, empieza la ejecución de este hilo principal, luego se define un `SurfaceView` que comunica el `Holder` de la cámara con el `Layout`, el cual será el contenedor de la cámara mientras la aplicación esté en uso.

Figura 4. *Mapa layouts.*



El acceso al BD para obtener todos los datos necesarios se realiza con la implementación de la clase `MyDataBase`, que extiende la clase `SQLiteAssetHelper`. Con la clase extendida se puede inicializar, acceder y manipular el BD `SQLite` local instalado en la aplicación con los datos previamente cargados. Para poder manipular el BD local, es necesario utilizar la carpeta `activos` de `Android` y crear un nuevo subdirectorio `databases`. Dentro de `MyDatabase` se harán todas las consultas necesarias para actualizar las listas globales de las entidades en tiempo de ejecución.

Una vez empezada la ejecución del hilo principal, a través de la interfaz `Runnable`, necesaria para la instancia de este hilo, se comunica con un nuevo hilo generado para la interfaz de usuario de la clase `UiThread`.

El método `onAcceleratorchanged`, permite obtener valores sobre el eje X del dispositivo en tiempo de ejecución. El valor obtenido sobre este eje informa el ángulo de giro y permite saber si el usuario sostiene de forma correcta dicho dispositivo. En caso contrario, la aplicación a través de una advertencia informará como debe mantenerse el dispositivo para que la funcionalidad sea la correcta.

El método procesarGraficosPDI recorre todos los PDI y aplica el teorema de la geodesia para obtener la posición de cada uno de ellos. Por cada PDI recorrido calcula el azimut teórico. Para calcular el azimut teórico obtiene la diferencia correspondiente entre la latitud y longitud cargada perteneciente al PDI seleccionado por el recorrido de la lista y la latitud y longitud de la ubicación actual. La ubicación actual se determina con la clase LocationManager. El uso de esta clase recibe notificaciones en cada actualización de la ubicación del dispositivo implementando el método onLocationChanged de LocationListener. La ubicación obtenida por este método devuelve la latitud y longitud del dispositivo.

Una vez restados ambos valores de las ubicaciones se obtiene el valor absoluto de estos y se dividen los valores, se calcula la tangente de esta división y se devuelve en grados. Con esta fórmula se obtiene el ángulo azimut teórico.

Es necesario establecer un límite inferior y otro superior. Cada PDI se mostrará si el resultado del cálculo anterior está dentro de los límites fijados al inicio de la aplicación y dentro de las configuraciones de usuario. Solo se visualiza un PDI por vez y si existieran varios PDI en una misma dirección, se visualiza el más cercano. Los PDI visualizados en pantalla tendrán los campos completados en forma dinámica.

La información corresponde a la tabla de los PDI del BD, el título corresponde al campo nombre. La categoría está asociada al id de categoría y de acuerdo con este id, la ventana emergente muestra un icono diferente (museos, patrimonio arquitectónico o patrimonio natural). La distancia se calcula con el algoritmo de Haversine, comparando la longitud y latitud del dispositivo con la longitud y latitud del PDI en el BD local de la aplicación.

Por último, la descripción de la ciudad, y un icono que representa el tipo de entrada (gratuita, de pago o a través de una donación). Esta información dentro del popup está diseñada con HTML con una de las propiedades del TextView de Android. Es un string concatenado que se procesa como HTML. En este tipo de propiedad no todos los tags de HTML están soportados y a través de la acción del usuario es posible presionar sobre este

popup emergente para obtener una nueva ventana del PDI, pero con mayor detalle e interacción.

El detalle está enfocado a los patrimonios que integran el PDI con sus respectivas imágenes y descripciones. Es posible presionar sobre un icono en forma de mano debajo de cada uno de los patrimonios del PDI y se registra la visita en el BD. El icono estará habilitado para presionarlo solo cuando los PDI se visualizan en tiempo de ejecución desde la actividad principal. Dentro de los buscadores no se dispone esta opción habilitada.

El administrador de la aplicación web puede ejecutar y visualizar gráficos. Estos informes sirven para visualizar los porcentajes de visitas a los diferentes PDI existentes en Paraguay. Para completar esta ventana, es necesario recorrer la tabla de patrimonios e imágenes asociados al PDI seleccionado por el usuario. Cada uno de los patrimonios se irán agregando a un ListView customizada instanciando la clase CustomListAdapter. Cuando el usuario presiona el icono de la mano, se realiza una llamada al Web Service (WS) para actualizar en el BD.

La llamada al WS se realiza a la uniform resource locator definida en la configuración del servidor (`{http://} + {IP}:{puerto} + {/entidad}`) indicando el método necesario para la consulta, obteniendo como respuesta un javascript object notation. Las clases de Android que invocan esta llamada al WS son `APIRestResourceGET` y `APIRestResourcePOST` para cada uno de los únicos métodos que se van a utilizar. Los datos recibidos se mapean con cada uno de los `ArrayList` globales definidos en la aplicación para actualizarlos en tiempo de ejecución.

Actividades de búsqueda

Las actividades secundarias se invocan desde la actividad principal por necesidad del usuario y las de búsquedas están implementadas utilizando la interface `SearchView.OnQueryTextListener` de Android. En esta aplicación se utiliza un widget de búsqueda, que permite escribir en la barra de menú la palabra a buscar y en cada cambio, el listener ejecuta el evento `onQueryTextChange`.

Este método filtra de acuerdo a lo escrito por el usuario y notifica al adaptador la búsqueda realizada y luego irá ocultando en el listado general las posiciones que no corresponden a la búsqueda en concreto.

Finalmente, desde la perspectiva del administrador, en ambiente web, se dispone de informaciones útiles respecto a la difusión y visitas a los patrimonios culturales e históricos registrados en línea.

Conclusiones

Se ha logrado la integración de tecnologías emergentes en una aplicación que permite brindar información acerca de la historia nacional centrada en lugares históricos de Paraguay y de distintos aspectos que hacen a su patrimonio cultural.

Mediante el uso de herramientas de realidad aumentada y algoritmos para la búsqueda de ubicaciones se ha conseguido detectar PDIs siguiendo una serie de pasos basados en la teoría de geodesia.

Incorpora con el uso de los sensores una innovadora experiencia de usuario que permite la orientación del dispositivo y posteriormente calcular las distancias entre diferentes puntos en un sistema de coordenadas geográficas de localización cercana con el algoritmo de haverine.

La utilización de esta nueva guía tecnológica, construido en un mapa layouts, combina en un escenario real contenido virtual, permite localizar y visualizar contenidos de forma simple y atractiva para usuarios de dispositivos móviles o tableta que funcionen con el sistema Android.

El ambiente web es gerenciado por el administrador de la base datos y gestor de los reportes generados por la aplicación.

Para investigaciones futuras que enriquezcan esta investigación, se recomienda introducir opciones multi-idioma a la aplicación, implementar algoritmos capaces de establecer rutas entre diferentes ubicaciones para realizar algún camino de interés particular bajo ciertos parámetros (Categoría de patrimonios, tipo de entrada, ciudades

cercanas, distancias más cortas), determinar una ruta en pantalla con RA capaz de mostrar el recorrido desde la ubicación actual hasta ese PDI, abrir la cantidad de canales de notificaciones incrementando la recepción de mensajes push de acuerdo a diferentes categorías.

Referencias

- Agencia De Información Paraguaya (2017). El 52% del patrimonio histórico del Paraguay se encuentra en el centro de Asunción. Recuperado de:
<https://www.ip.gov.py/ip/el-centro-historico-de-asuncion-cuenta-con-el-52-del-patrimonio-del-paraguay/>.
- ANDROID Y GOOGLE (2017). Guía de usuario. Recuperado de:
<https://developer.android.com/studio/intro/index.html?hl=es-419>
- Azuma. R. (1997). A Survey of Augmented Reality. Presence: Teleoperators and Virtual Environments 6, 4 (August 1997), 355-385. Recuperado de:
<http://ronaldazuma.com/papers/ARpresence.pdf>.
- Bonsor, K. (2011). How Augmented Reality Works: Limitations and the Future of Augmented Reality. Recuperado de:
<https://computer.howstuffworks.com/augmented-reality4.htm>
- Lens-Fitzgerald, M. (2009). Augmented Reality Hype Cycle. SPRXmobile: Mobile Service Architects. Recuperado de: <http://www.sprxmobile.com/the-augmented-reality-hype-cycle/>.
- Ley N° 5621 De Protección Del Patrimonio Cultural(2016). Gaceta Oficial de la República del Paraguay. Asunción, Paraguay, 12 de julio de 2016.
- Milgram, P. Y Kishino, F. (1994). A Taxonomy of Mixed Reality Visual Displays. IEICE Transactions on Information and Systems, vol E77-D, 1321-1329. Recuperado de: <http://www.alice.id.tue.nl/references/milgram-kishino-1994.pdf>.
- Morcillo, C., Vallejo, D., Albusac, J. Y Castro, J. (2013). Realidad Aumentada. Un enfoque práctico con ARToolKit y Blender. España: Bubok Publishing S.L.
- MOVABLE TYPE SCRIPTS (2019). Calculate distance, bearing and more between Latitude/Longitude points. Recuperado el 23 de mayo de 2019 de:

<https://www.movable-type.co.uk/scripts/latlong.html>

Observatorio Turístico Nacional (s.f.). Ranking de turistas a Paraguay, por año, según nacionalidad. Recuperado de:

http://www.observatorionacionaldeturismo.gov.py/application/files/3714/6841/9606/ranking_de_visitantes_a_paraguay_por_nacionalidad_periodo_2010_a_2015_-_copia_2.pdf.

Prendes, C. (2015). Realidad aumentada y educación: análisis de experiencias prácticas. *Píxel-Bit. Revista de Medios y Educación*, 46, 187-203.

PYSHICS (2019). What is reason for electronic compass calibration?. Recuperado de: <https://physics.stackexchange.com/questions/17587/what-is-reason-for-electronic-compass-calibration>.

Tomaselli, J. (2019). Realidad aumentada y geolocalización cercana destinadas al patrimonio cultural de Paraguay. Trabajo de conclusión de carrera. Carrera de Ingeniería en Informática, Universidad Columbia del Paraguay.

UNESCO (1982). Conferencia mundial sobre las políticas culturales, 2nd., Mexico City, 1982. Recuperado de https://unesdoc.unesco.org/ark:/48223/pf0000052505_spa.