

Mapas do Tour por París

Neste capítulo, vai construír unha aplicación que permite que cree a súa propio guía personalizada para unha viaxe de soño en París. E xa que algúns dos teus amigos non poden unirse a ti, imos crear unha aplicación que lles permita facer un tour virtual de París tamén. Crear unha aplicación de mapa en pleno funcionamento pode parecer moi complicado, pero o App Inventor permite que use o compoñente ActivityStarter para lanzar o Google Maps para cada lugar virtual. Primeiro, vai desenvolver unha aplicación que lanza mapas para a Torre Eiffel, o Louvre e Catedral de Notre Dame, con un só click. Entón vai modificar a aplicación para crear un tour virtual de mapas de satélite que tamén están dispoñibles a partir de Google.

O que aprenderás

Este capítulo presenta os seguintes compoñentes App Inventor e conceptos relacionados:

- O compoñente de inicio de Actividade **ActivityStarter** para o lanzamento doutras aplicacións de Android desde a súa aplicación. Vai usar este compoñente aquí para lanzar Google Maps con varios parámetros.
- O compoñente ListPicker para permitir que o usuario escolla a partir dunha lista de lugares.

Deseñando os compoñentes

Crea un novo proxecto no App Inventor e chámalo "ParisMapTour". A interface de usuario para a aplicación ten un compoñente de imaxe con unha foto de París, un compoñente Label con algún texto, un compoñente ListPicker que ven con un botón asociado, e un compoñente ActivityStarter (non visible). Pode proxectar os compoñentes usando a imaxe da Figura 6-1. Os compoñentes listados na Táboa 6-1 foron utilizados para crear esta xanela Deseño. Arrastra cada compoñente da Paleta ao Viewer e noméaos como se indica.



Imaxe 6-1. A aplicación Mapas do Tour de París en execución no emulador

Táboa 6-1. Compoñentes para a aplicación Mapas do Tour de París

Component type	Palette group	What you'll name it	Purpose
Image	Basic	Image1	Show a static image of a Paris map on screen.
Label	Basic	Label1	Display the text "Discover Paris with your Android!"
ListPicker	Basic	ListPicker1	Display the list of destination choices.
ActivityStarter	Other stuff	ActivityStarter1	Launch the Maps app when a destination is chosen.

Definindo as propiedades ActivityStarter

ActivityStarter é un compoñente que permite que abrir calquera aplicación dun navegador Android, Google Maps, ou incluso un dos seus propios programas. Cando un usuario inicia outra app desde a súa aplicación, el poderá premer no botón back-voltar para voltar á súa aplicación. Vai construír ParisMapTour para que a aplicación Mapas sexa lanzada para mostrar mapas específicos con base na elección do usuario. O usuario pode, entón, acertar o botón back-voltar para voltar ao seu programa e elixir un destino diferente.

ActivityStarter é un compoñente de baixo nivel en concepto no que ten que definir algunhas propiedades con información familiar para un programador Java SDK Android, pero estraños para os outros 99,999% do mundo. Para esta aplicación, introduza as propiedades conforme especificados na táboa 6-2, e teña coidado, xa que poñer as letras en maiúsculas ou minúsculas é algo importante.

Táboa 6-2. propiedades de ActivityStarter para lanzar Google Maps

Property	Value
Action	android.intent.action.VIEW
ActivityClass	com.google.android.maps.MapActivity
ActivityPackage	com.google.android.apps.maps

No editor de bloques, vai establecer unha propiedade, DataUri, que permite lanzar un mapa específico en Google Maps. Esta propiedade debe ser definida no editor de bloques en vez da deseñador de compoñentes, porque ten que ser dinámico; que vai cambiar, dependendo se o usuario decide visitar a Torre Eiffel, o Louvre, ou a Catedral de Notre Dame.

Nós imos chegar ao editor de bloques en só un momento, pero hai un par de detalles para tomar coidado antes de lle pode pasar á programación do comportamento para os seus compoñentes:

1. Baixe o arquivo aquí metro.jpg metro.jpg para cargar no seu proxecto. Logo, necesitará configúrelo como a propiedade Picture de Image1.
2. O compoñente ListPicker ven con un botón; cando o usuario fai clic nel, preséntanse as opcións. Define o texto do botón, cambiando a propiedade text de ListPicker1 a "escolle París Destino".

Adición de comportamentos aos Compoñentes

No editor de bloques, ten que definir unha lista de destinos e dous comportamentos:

- Cando a aplicación comeza, a aplicación leva os destinos no compoñente ListPicker que o usuario poida elixir un.
- Cando o usuario escolle un destino de ListPicker, a aplicación Mapas é lanzada e mostra un mapa deste destino. Nesta primeira versión da aplicación, só vai abrir a aplicación Mapas e dicirlle que realice unha busca ao destino elixido.

Creando unha lista de destinos

Abre o editor de bloques e crear unha variable coa lista de destinos de París, usando os bloques listados na Táboa 6-3.

Táboa 6-3. Bloques para a creación dunha variable destinos

Block type	Drawer	Purpose
def variable ("Destinations")	Definitions	Create a list of the destinations.
make a list	Lists	Add the items to the list.
text ("Tour Eiffel")	Text	The first destination.
text ("Musée du Louvre")	Text	The second destination.
text ("Cathédrale Notre Dame")	Text	The third destination.

A variable *destinations* chamará a facer unha función lista, na cal pode conectar os valores de texto para os tres destinos na súa xira, como se mostra na Figura 6-2.

Imaxe 6-2. A creación dunha lista é doado no App Inventor



Permitindo que o usuario seleccione un destino

O obxectivo do compoñente ListPicker é amosar unha lista de elementos para que o usuario escolla. Vostede precargará as opcións para o ListPicker definindo os elementos de propiedade a unha lista. Para esta aplicación, quere establecer a propiedade Elements da ListPicker igual á lista de destinos que acaba de crear. Porque quere ver a lista cando a aplicación sexa iniciada, vai configurar este comportamento no evento Screen1.Initialize. Vai ter os bloques listados na Táboa 6-4.

Block type	Drawer	Purpose
Screen1.Initialize	Screen1	This event is triggered when the app starts.
set ListPicker1 .Elements to	ListPicker1	Set this property to the list you want to appear.
global destinations	My Definitions	The list of destinations.

Táboa 6-4. Bloques para o lanzamento do ListPicker cando a aplicación se executa

Como traballan os bloques

Screen1.Initialize é disparado cando a aplicación comeza. Como se mostra na Figura 6-3, o procesador de eventos define a propiedade Elements de ListPicker de xeito que os tres destinos aparecen.



Imaxe 6-3. Poña calquera cousa que queira que aconteza cando a aplicación comece nun manipulador de eventos Screen1.Initialize



Probe a súa apps. Primeiro, necesitará reiniciar a aplicación, premendo en "Entrar ao

dispositivo ..." no editor de bloques. A continuación, no teléfono, prema no botón "Seleccionar París Destino". A lista seleccionados debe aparecer cos tres elementos.

Abrindo Maps unha investigación

Logo vai programar a aplicación para que cando o usuario escolla un dos destinos, o ActivityStarter lance Google e procure o lugar seleccionado.

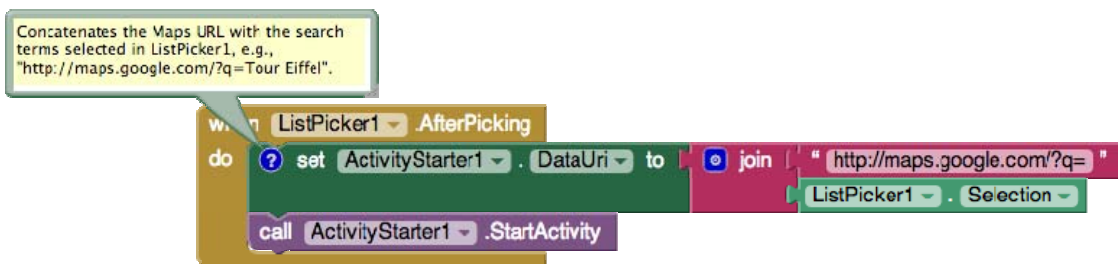
Cando o usuario elixe un elemento do compoñente ListPicker, o evento ListPicker.AfterPicking é accionado. O manipulador de eventos para AfterPicking, ten que definir o DataUri do compoñente ActivityStarter para que saiba que o mapa abrir e, a continuación, ten que iniciar a Google Maps usando ActivityStarter.StartActivity. Os bloques para esta función están listados na Táboa 6-5.

Táboa 6-5. Bloques para lanzar Google Maps coa actividade de inicio

Block type	Drawer	Purpose
ListPicker1.After Picking	ListPicker1	This event is triggered when the user chooses from ListPicker.
set ActivityStarter1 .DataUri to	ActivityStarter1	The DataUri tells Maps which map to open on launch.
make text	Text	Build the DataUri from two pieces of text.
text ("geo:0,0?q=")	Text	The first part of the DataUri expected by Maps.
ListPicker1.Selection	ListPicker1	The item the user chose.
ActivityStarter1 .StartActivity	ActivityStarter1	Launch Maps.

Como traballan os bloques

Cando o usuario escolle a partir do ListPicker, o elemento seleccionado é almacenado en ListPicker.Selection e o evento AfterPicking é disparado. Como se mostra na Figura 6-4, a propiedade DataUri defínese como un obxecto de texto que combina "http://maps.google.com/?q =" co produto escollido. Así, se o usuario escolleu o primeiro elemento, "Tour Eiffel", o DataUri sería definida como "http://maps.google.com/?q=Tour Eiffel."



Imaxe 6-4. Definir o DataURI para lanzar o mapa seleccionado

Dende que xa estableceu as outras propiedades da ActivityStarter para que saiba abrir Maps, o bloque ActivityStarter1.StartActivity lanza a aplicación Mapas e invoca a procura prescrita pola DataUri.



Probe a súa aplicación. Reinicie a aplicación e prema de novo o botón "Escoller Destino París ". Cando escolle un dos destinos, é que un mapa de destino que aparece? Google Maps tamén

debe proporcionar un botón atrás para volver para que a súa aplicación poda escoller de novo, isto funciona? (Pode ter que facer clic no botón atrás un par de veces.)

Configurar un Tour Virtual

Agora imos darlle salsa á aplicación e buscar algúns puntos de vista dos monumentos de París para que os seus amigos na casa podan seguirte mentres estás fóra. Para iso, primeiro explore no Google Maps os URLs dalgúns mapas específicos. Aínda vai usar os mesmos marcos parisienses para os destinos, pero cando o usuario escolle un, vai usar o índice (a posición na lista) da súa elección para seleccionar e abrir un zoom-in específico ou un mapa tipo StreetView.

Antes de ir, pode querer gardar o seu proxecto (usando Gardar como) para que teña unha copia do mapa turístico sinxelo que creou ata agora. Desta forma, se fai algo que causa problemas na súa aplicación, sempre pode volver a esta versión a traballar e tentalo de novo.

Buscar o DataUri para mapas específicos

O primeiro paso é abrir o Google Maps no seu ordenador para atopar os mapas específicos que quere publicar en cada destino:

1. No seu ordenador, desprácese ata <http://maps.google.com>.
2. Buscar un marco (por exemplo, a Torre Eiffel).
3. Zoom para o nivel que quere.
4. Escolla o tipo de vista que quere (por exemplo, enderezo, Satélite ou Street View).
5. Prema no botón ligazón na parte superior dereita da xanela de Google Maps e copia a URL para o mapa. Vai usar esta URL (ou partes dela) para lanzar o mapa na súa aplicación. Empregando este esquema a Táboa 6-6 mostra as URL que vai empregar.

Táboa 6-6. Tour virtual da URL a Google Maps

Landmark	Maps URL
Tour Eiffel	http://maps.google.com/maps?f=q&source=s_q&hl=en&geocode=&q=eiffel+tower&sl=37.0625,-95.677068&sspn=48.909425,72.333984&ie=UTF8&hq=Tour+Eiffel&hnear=Tour+Eiffel,+Quai+Branly,+75007+Paris,+Ile-de-France,+France&ll=48.857942,2.294748&spn=0.001249,0.002207&t=h&z=19
Musée du Louvre	http://maps.google.com/maps?f=q&source=s_q&hl=en&q=louvre&sl=48.86096,2.335421&sspn=0.002499,0.004415&ie=UTF8&t=h&split=1&filter=0&rq=1&ev=zi&radius=0.12&hq=louvre&hnear=&ll=48.86096,2.335421&spn=0.002499,0.004415&z=18
Cathédrale Notre Dame (Street View)	http://maps.google.com/maps?f=q&source=s_q&hl=en&q=french+landmarks&sl=48.853252,2.349111&sspn=0.002411,0.004415&ie=UTF8&t=h&radius=0.12&split=1&filter=0&rq=1&ev=zi&hq=french+landmarks&hnear=&ll=48.853252,2.349111&spn=0.004415&z=18&layer=c&cbll=48.853046,2.348861&panoid=74fLTqeYdgkPYj6KKLlqgQ&cbp=12,63.75,,0,-35.58

Para ver calquera destes mapas, pegue as URLs da Táboa 6-6 nun navegador. Os dous primeiros mapas son o zoom-in imaxes de satélite, mentres que o terceiro é unha vista de rúa. Podes usar estas URLs directamente para lanzar os mapas que queres, ou podes definir URLs limpas, utilizando os protocolos descritos en Google Maps <http://mapki.com>. Por exemplo, pode amosar o mapa Torre Eiffel usando só as coordenadas GPS atopadas na URL longa na Táboa 6-6 e facer un mapa con protocolo geo:

geo: 48.857942,2.294748 t = h & z = 19

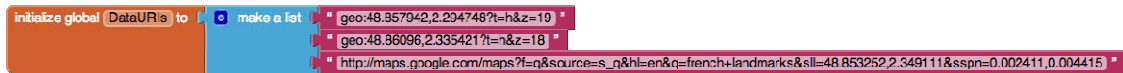
Usando tal DataUri, vai ter esencialmente o mesmo mapa que o mapa con base na URL completa a partir do cal foron extraídas as coordenadas GPS. A t = h especifica que mapas deben mostrar un mapa híbrido con satélite e abordar puntos de vista, e z = 19 especifica o

nivel de zoom. Se vostede está interesado nos detalles de definición de parámetros para varios tipos de mapas, busque a documentación no <http://mapki.com>.

Para sentirse cómodo usando os dous tipos de URLs, usaremos o formato geo: para as dúas primeiras definicións DataUri da nosa lista, e a URL completa para o terceiro.

Establecer a lista de dataURIs

Vai ter unha lista nomeada dataURIs, contendo unha DataURI para cada mapa que quere amosar. Crear esta lista, como se mostra na Figura 6-5 para que os elementos corresponden aos elementos da lista de destinos (ou sexa, o primeiro, dataURI debe corresponder ao primeiro destino, a Torre Eiffel).



Imaxe 6-5. A lista de mapas para o tour virtual

Os dous primeiros elementos son presentados DataURIs á Torre Eiffel e o Louvre. Ambos usan o protocolo geo:. O terceiro DataURI non se mostra totalmente xa que o bloque é moi longo para esta páxina; ten que copiar este URL a partir da entrada de "Notre Dame, Street View" na táboa 6-6 e poñelas nun bloque de texto.

Modificando o comportamento de ListPicker.AfterPicking

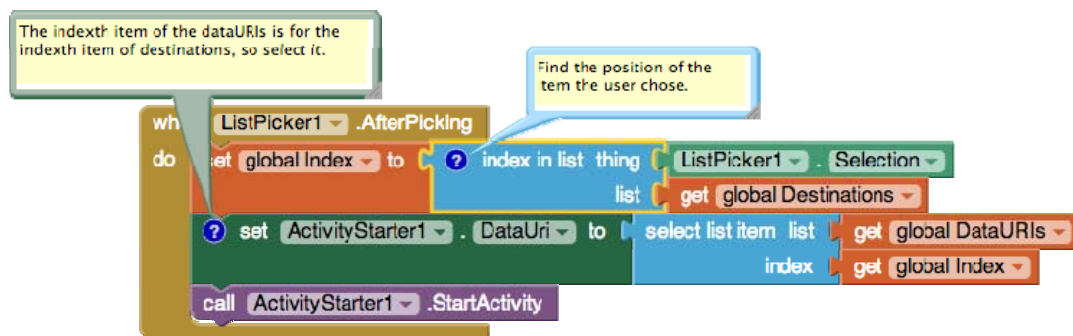
Na primeira versión deste programa, o comportamento de ListPicker.AfterPicking establece o DataUri como a concatenación (ou combinación) de "http://maps.google.com/?q =" e o destino que o usuario escolleu a partir da lista (por exemplo, Tour Eiffel). Nesta segunda versión, o comportamento AfterPicking debe ser máis sofisticado, pois o usuario está escollendo dunha lista (destinos), pero o DataUri debe ser seleccionado a partir doutra lista (dataURIs). En concreto, cando o usuario escolle un elemento da ListPicker, ten que saber que o índice da súa elección para que poida usalo para seleccionar o DataUri correcto na lista dataURIs. Imos explicar máis sobre o que é un índice en un momento, pero axuda a configurar o bloques primeiro a ilustrar mellor o concepto. Son moi poucos os bloques necesarios para esa función, os cales están listados na Táboa 6-7.

Táboa 6-7. Bloques para seleccionar un elemento da lista con base na selección do usuario

Block type	Drawer	Purpose
def variable ("index")	Definitions	This variable will hold the index of the user's choice.
number (1)	Math	Initialize the index variable to 1.
ListPicker1 .AfterPicking	ListPicker1	This event is triggered when the user chooses an item.
set global index to	My Definitions	Set this variable to the position of the selected item.
position in list	Lists	Get the position (index) of a selected item.
ListPicker1 .Selection	ListPicker1	The selected item—for example, "Tour Eiffel." Plug this into the "thing" slot of position in list .
global destinations	My Definitions	Plug this into the "list" slot of position in list .
set ActivityStarter .DataUri	ActivityStarter	Set this before starting the activity to open the map.
select list item	Lists	Select an item from the dataURIs list.
global DataURIs	My Definitions	The list of DataURIs.

Como traballan os bloques

Cando o usuario escolle un elemento da ListPicker, o evento AfterPicking é accionado, como se mostra na Figura 6-6. O elemento -por exemplo- elixido, "Tour Eiffel", está na ListPicker.Selection. O manipulador de eventos usa isto para atopar a posición do elemento seleccionado, ou o valor do índice, na lista destinos. O índice corresponde á posición do destino seleccionado na lista. Entón, se "Tour Eiffel" é seleccionado, o índice será 1; se "Musée du Louvre" é seleccionado, será 2; e se "Catedral de Notre Dame de París" é seleccionado, o índice será de 3.



Imaxe 6-6. A elección dun elemento da lista con base na selección do usuario. O índice pode entón ser usado para seleccionar un elemento doutra lista, neste caso, dataURIs-e para definir esta entrada como DataUri do ActivityStarter. Unha vez feito iso, o mapa pode ser lanzado con ActivityStarter.StartActivity.



Probe a súa aplicación. No teléfono, prema no botón "Escoller Destino París ". Debe aparecer unha lista cos tres elementos. Escolla un dos elementos e vexa o mapa que aparece.

variacións

Aquí están algunhas variacións que lle suxerimos facer:

- Crear un tour virtual de algún outro destino exótico, ou do seu lugar de traballo ou na escola.
- Crear unha aplicación Tour virtual personalizable que permita ao usuario crear unha guía para un lugar da súa elección, escribindo o nome de cada destino, xunto coa URL dun mapa correspondente. Vai ter que almacenar os datos nunha base de datos TinyWebDB e crear unha aplicación Tour virtual que traballa cos datos introducidos. Para un exemplo de como crear unha base de datos TinyWebDB, consulta a aplicación MakeQuiz / TakeQuiz.

resumo

Aquí están algunhas das ideas que se cobren neste capítulo:

- Variables de lista poden usarse para almacenar datos como destinos de mapa e URLs
- O compoñente **ListPicker** permite ao usuario escoller entre unha lista de elementos. A propiedade **Elements** do **ListPicker** contén a lista, a propiedade **Selection** contén o elemento seleccionado, e o evento **AfterPicking** é accionado cando o usuario escolle un elemento da lista.

- O compoñente ActivityStarter permítelle a súa aplicación lanzar outras aplicacións. Este capítulo demostrou o seu uso coa aplicación Maps, pero pode lanzar un navegador ou calquera outra aplicación Android, así como, incluso outra que mesmo creou vostede. Vexa <http://appinventor.googlelabs.com/learn/reference/other/activitystarter.html> para máis información.
- Pode iniciar un mapa especial en Google Maps, que define a propiedade DataUri. Podes atopar URI, configurando un mapa especial no navegador e, a continuación, seleccionando o botón ligazón para atopar o URI. Podes poñer un URI como directamente no DataUri da súa ActivityStarter, ou construír a súa propia URI utilizando os protocolos establecidos no <http://mapki.com>.
- Pode identificar o contido dun elemento da lista a través da posición no bloque lista. Con ListPicker, pode utilizar a posición da lista para atopar o contido do elemento que o usuario escolle. Isto é importante cando, como neste capítulo, depende do índice para escoller un elemento a partir dunha segunda lista, relacionada. Para máis información sobre variables de lista e o compoñente ListPicker, consulte outros titoriais.

Bibliografía Para a versión 1 de App Inventor

<http://s445852199.mialojamiento.es/peponees/appinventorspanish/learn/tutorials/maptour/maptour.html>

<http://www.appinventor.org/apps/map-tour/map-tour.pdf>

<http://beta.appinventor.mit.edu/learn/tutorials/maptour/maptour.html>

<http://beta.appinventor.mit.edu/learn/tutorials/maptour/maptour2.html>