
globsyn
Taking People To The Next Level

globsyn


globsyn
finishing school

Sharing Data

Sharing Data

- Topics to be covered in this session:
  - Sharing data

Sharing Data

Android applications can communicate and integrate with each other. 3 Steps involved are:

- **Sending Content to Other Apps:**
  Learn how to set up your application to be able to send text and binary data to other applications with intents.

- **Receiving Content from Other Apps:**
  Learn how to set up your application to receive text and binary data from intents.

- **Adding an Easy Share Action:**
  Learn how to add a "share" action item to your action bar.

Sharing Data Contd.

- Sending Content to Other Apps:

  - Using an intent, you must specify the action you want the intent to "trigger."
  - Android defines several actions ex: ACTION_SEND indicates that the intent is sending data from one activity to another, even across process boundaries.
  - Need to do is specify the data and its type to another activity, the system will identify compatible receiving activities and display them to the user or immediately start the activity.
  - Intents allow users to share information quickly and easily, using their favorite applications.

Sharing Data Contd.

- 3 ways to send content to other applications using intents are:
  - **Send text Content:**

Use the ACTION_SEND action to send text content from one activity to another.

Ex: The built-in Browser app can share the URL of the currently-displayed page as text with any application. Useful for sharing an article or website with friends via email or social networking.

  - **Send Binary Content:**

Binary data is shared using the ACTION_SEND action combined with setting the appropriate MIME type and placing the URI to the data in an extra named EXTRA_STREAM. Ex: To share an image or any type of binary content:

Sharing Data Contd.

- **Send Multiple pieces of data:**

  - To share multiple pieces of content, use the ACTION_SEND_MULTIPLE action together with a list of URIs pointing to the content. The MIME type varies according to the mix of content you're sharing.

Ex: you share 3 JPEG images, the type is still "image/jpeg". For a mixture of image types, it should be "image/*" to match an activity that handles any type of image.

  - Use "*/*" if you're sharing out a wide variety of types.

  - It's up to the receiving application to parse and process your data.

Sharing Data Contd.

The second step in Sharing data is Receiving Content from Other Apps. It can be done by:

- Update your manifest:
  - Intent filters inform the system what intents an application component is willing to accept.
  - Create intent filters in order to be able to receive intents with this action.
  - Define an intent filter in your manifest, using the <intent-filter> element.

Sharing Data Contd.

Handle the Incoming Content:

- Start by calling getIntent() to get Intent object.
- Once you have the object, you can examine its contents to determine what to do next.
- Need to check if this activity can be started from other parts of the system, such as the launcher when examining the intent.

Sharing Data Contd.

The third step in Sharing data is Adding an Easy Share Action:

- ActionProvider makes implementing an effective and user friendly share action in your ActionBar.

- Once attached to a menu item in the action bar, it handles both the appearance and behavior of that item. In the case of ShareActionProvider, you provide a share intent and it does the rest.

Sharing Data Contd.

Update Menu Declarations:

- To use ShareActionProviders, define the android:actionProviderClass attribute for the corresponding <item> in your menu resource file:

- This delegates responsibility for the item's appearance and function to ShareActionProvider. However, you will need to tell the provider what you would like to share.

Sharing Data Contd.

- Set the share intent:
  - For ShareActionProvider to function, you must provide it a share intent. This share intent should be the same as described in the Sending Content to Other Apps, with action ACTION_SEND and additional data set via extras like EXTRA_TEXT and EXTRA_STREAM.
  - To assign a share intent, find the corresponding MenuItem while inflating your menu resource in your Activity or Fragment.
  - Call MenuItem.getActionProvider() to retreive an instance of ShareActionProvider.

Sharing Data Contd.

- Use setShareIntent() to update the share intent associated with that action item

- Can either need to set the share intent once during the creation of your menus, or may want to set it and then update it as the UI changes.

References

- http://developer.android.com/training/sharing/index.html


Taking People To The Next Level ...