Android file chooser [closed]

Asked 7 years, 10 months ago   Active 4 years, 7 months ago   Viewed 199k times

▲

108

▼

I want to make a file uploader. And I hence I need a file chooser but I don't want to write this by myself. I find OI file manager and I think it suits me. But how can I force user to install OI file manager? If I cannot , is there a better way to include a file manager in my app? Thx

★

70

| android | file-upload | file-io | filechooser | android-afilechooser |

edited Nov 25 '14 at 5:25   asked Oct 22 '11 at 2:08

Pratik Butani   Bear
AndroidDev   **2,488**   4   43   68
33.4k   28   155   293

---

closed as too broad by Andrew Barber Oct 29 '13 at 21:19

Please edit the question to limit it to a specific problem with enough detail to identify an adequate answer. Avoid asking multiple distinct questions at once. See the How to Ask page for help clarifying this question.

If this question can be reworded to fit the rules in the help center, please edit the question.

I use github.com/18446744073709551615/android-file-chooser-dialog – 18446744073709551615 Jul 24 '14 at 13:01

2 Answers

▲

255

▼

✓

EDIT (*02 Jan 2012*):

I created a small open source Android Library Project that streamlines this process, while also providing a **built-in file explorer** (in case the user does not have one present). It's extremely simple to use, requiring only a few lines of code.

You can find it at GitHub: aFileChooser.

ORIGINAL

If you want the user to be able to choose any file in the system, you will need to include your own file manager, or advise the user to download one. I believe the best you can do is look for "openable" content in an `Intent.createChooser()` like this:

```
private static final int FILE_SELECT_CODE = 0;

private void showFileChooser() {
    Intent intent = new Intent(Intent.ACTION_GET_CONTENT);
    intent.setType("*/*");
    intent.addCategory(Intent.CATEGORY_OPENABLE);

    try {
        startActivityForResult(
```

```
                Intent.createChooser(intent, "Select a File to Upload"),
                FILE_SELECT_CODE);
    } catch (android.content.ActivityNotFoundException ex) {
        // Potentially direct the user to the Market with a Dialog
        Toast.makeText(this, "Please install a File Manager.",
                Toast.LENGTH_SHORT).show();
    }
}
```

You would then listen for the selected file's `Uri` in `onActivityResult()` like so:

```
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    switch (requestCode) {
        case FILE_SELECT_CODE:
        if (resultCode == RESULT_OK) {
            // Get the Uri of the selected file
            Uri uri = data.getData();
            Log.d(TAG, "File Uri: " + uri.toString());
            // Get the path
            String path = FileUtils.getPath(this, uri);
            Log.d(TAG, "File Path: " + path);
            // Get the file instance
            // File file = new File(path);
            // Initiate the upload
        }
        break;
    }
    super.onActivityResult(requestCode, resultCode, data);
}
```

The `getPath()` method in my `FileUtils.java` is:

```
public static String getPath(Context context, Uri uri) throws URISyntaxException
{
    if ("content".equalsIgnoreCase(uri.getScheme())) {
        String[] projection = { "_data" };
        Cursor cursor = null;

        try {
            cursor = context.getContentResolver().query(uri, projection, null,
null, null);
            int column_index = cursor.getColumnIndexOrThrow("_data");
            if (cursor.moveToFirst()) {
                return cursor.getString(column_index);
            }
        } catch (Exception e) {
            // Eat it
        }
    }
    else if ("file".equalsIgnoreCase(uri.getScheme())) {
        return uri.getPath();
    }

    return null;
}
```

edited Jun 5 '13 at 22:46	answered Oct 22 '11 at 2:49
Xavi	Paul Burke
16.3k 12 65 60	**23.7k** 8 59 60

---

1   I'd even go so far as to also direct them to app on the market. – Kurtis Nusbaum Oct 22 '11 at 2:50

---

1   Thx, but I get the content uri ---

content://org.openintents.cmfilemanager/mimetype//mnt/sdcard/1318638826728.jpg instead of file path. Could I convert it into file path? – Bear Oct 22 '11 at 5:14 ✎

2   But I can't find FileUtils.... – Bear Oct 22 '11 at 5:42 ✎

2   @Bicou Thanks for the note. Your nudge helped me to stop being lazy and make some minor changes. :-) I just pushed an update to the library that includes the license. – Paul Burke Sep 4 '12 at 6:15

19   This answer does not consider the uri like this:"content://com.android.providers.media.documents/document/image:62". – wangqi060934 Jul 31 '15 at 7:38

▲

-3

▼

I used AndExplorer for this purpose and my solution is popup a dialog and then redirect on the market to install the misssing application:

My startCreation is trying to call external file/directory picker. If it is missing call show installResultMessage function.

```java
private void startCreation(){
    Intent intent = new Intent();
    intent.setAction(Intent.ACTION_PICK);
    Uri startDir = Uri.fromFile(new File("/sdcard"));

    intent.setDataAndType(startDir,
            "vnd.android.cursor.dir/lysesoft.andexplorer.file");
    intent.putExtra("browser_filter_extension_whitelist", "*.csv");
    intent.putExtra("explorer_title",
 getText(R.string.andex_file_selection_title));
    intent.putExtra("browser_title_background_color",
            getText(R.string.browser_title_background_color));
    intent.putExtra("browser_title_foreground_color",
            getText(R.string.browser_title_foreground_color));
    intent.putExtra("browser_list_background_color",
            getText(R.string.browser_list_background_color));
    intent.putExtra("browser_list_fontscale", "120%");
    intent.putExtra("browser_list_layout", "2");

    try{
        ApplicationInfo info = getPackageManager()
                            .getApplicationInfo("lysesoft.andexplorer", 0
);

        startActivityForResult(intent, PICK_REQUEST_CODE);
    } catch( PackageManager.NameNotFoundException e ){
        showInstallResultMessage(R.string.error_install_andexplorer);
    } catch (Exception e) {
        Log.w(TAG, e.getMessage());
    }
}
```

This methos is just pick up a dialog and if user wants install the external application from market

```java
private void showInstallResultMessage(int msg_id) {
    AlertDialog dialog = new AlertDialog.Builder(this).create();
    dialog.setMessage(getText(msg_id));
    dialog.setButton(getText(R.string.button_ok),
            new DialogInterface.OnClickListener() {

                @Override
                public void onClick(DialogInterface dialog, int which) {
                    finish();
                }
            });
```

```java
        dialog.setButton2(getText(R.string.button_install),
                new DialogInterface.OnClickListener() {

                    @Override
                    public void onClick(DialogInterface dialog, int which) {
                        Intent intent = new Intent(Intent.ACTION_VIEW);
                        intent.setData(Uri.parse("market://details?
id=lysesoft.andexplorer"));
                        startActivity(intent);
                        finish();
                    }
                });
        dialog.show();
    }
```

answered Oct 24 '11 at 6:08

andep
6,855　1　9　4

So, one of the requirements for your app users is... install AndExplorer?! – Fantômas Mar 14 '14 at 13:05