

To make Dictan available for the external applications, it should be initialized, i.e. the dictionary base should be selected. The last selected base will be used to view articles for the external applications.

The CallerActivity below shows an example of the Dictan call. We assume that the layout 'R.layout.main' contains a button with the attribute 'android:onClick="testDictanCall"'. The method to view the article is 'showDictanArticle(String word)' which accepts the word for the article. In other words, if we want to translate the word "butter", the method call is 'showDictanArticle("butter")'.

After Dictan is requested, the result comes to the method 'onActivityResult(int requestCode, int resultCode, Intent intent)' which contains the status and error message if it occurred.

```
import android.app.Activity; import android.content.Intent; import android.os.Bundle; import
android.view.View; import android.widget.Toast; /** * * @since version 3.0, 12/26/2011 *
* @modified version 3.5, 11/23/2013 * @modified version 4.7, 04/12/2015 * * @author
Dmitry Viktorov */ public class CallerActivity extends Activity { private final static int
DICTAN_ARTICLE_REQUEST_CODE = 100; private final static String
DICTAN_ARTICLE_WORD = "article.word"; private final static String
DICTAN_ERROR_MESSAGE = "error.message"; @Override public void
onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState);
setContentView(R.layout.activity_caller); } public void testDictanCall(View view) {
showDictanArticle("butter"); } public void testDictanCallWithNonExistentWord(View
view) { showDictanArticle("non_existent_word"); } public void
showDictanArticle(String word) { try { Intent intent = new
Intent(Intent.ACTION_VIEW); // Add custom category to run the Dictan
external dispatcher
intent.addCategory("info.softex.dictan.EXTERNAL_DISPATCHER"); // Don't
include the dispatcher in activity in history // because it doesn't have any content view.
intent.addFlags(Intent.FLAG_ACTIVITY_NO_HISTORY); // Don't use transition
animation intent.addFlags(Intent.FLAG_ACTIVITY_NO_ANIMATION);
intent.putExtra(DICTAN_ARTICLE_WORD, word); startActivityForResult(intent,
DICTAN_ARTICLE_REQUEST_CODE); // Intent.FLAG_ACTIVITY_NO_ANIMATION
is sometimes ignored // Explicitly disable the pending transition
overridePendingTransition(0, 0); } catch (Exception e) {
Toast.makeText(getApplicationContext(), "There was a problem loading Dictan: " +
e.getMessage(), Toast.LENGTH_LONG).show(); } } @Override
protected void onActivityResult(int requestCode, int resultCode, Intent intent) { if
(requestCode == DICTAN_ARTICLE_REQUEST_CODE) { switch (resultCode) {
// The article has been shown, the intent is never expected to be null case
```

```
RESULT_OK:    String okMessage = "The Requested Word: " +
intent.getStringExtra(DICTAN_ARTICLE_WORD) + ". The article has been shown.";
Toast.makeText(getApplicationContext(), okMessage,    Toast.LENGTH_LONG).show();
break;        // Error occurred  case RESULT_CANCELED:    String errMessage = "Unknown
Error.";    if (intent != null) {    errMessage = "The Requested Word: " +
intent.getStringExtra(DICTAN_ARTICLE_WORD) + ". Error: " +
intent.getStringExtra(DICTAN_ERROR_MESSAGE);    }
Toast.makeText(getApplicationContext(), errMessage,    Toast.LENGTH_LONG).show();
break;        // Must never occur  default:    Toast.makeText(getApplicationContext(),
"Unknown Result Code: " + resultCode,    Toast.LENGTH_LONG).show();  break;    }
}  }
```