Zello SDK Migration Guide

Major changes have been made to the new version of the Android Zello SDK. These changes are not backwards compatible. However, the necessary migration is fairly straightforward.

The main changes are:

- The *Sdk* class has been replaced with a static *Zello* class. If your previous implementation look something like this:

```
private com.zello.sdk.Sdk sdk = new com.zello.sdk.Sdk();
private void startVoiceMessage() {
   MyActivity.this.sdk.beginMessage();
}
```

  It would be replaced with this:

```
private void startVoiceMessage() {
   Zello.beginMessage();
}
```

- The lifecycle methods have loosely translated in the following way:
  - Sdk.this.onCreate() -> Zello.initialize()
  - Sdk.this.onResume() -> Zello.leavePowerSavingMode()
  - Sdk.this.onPause() -> Zello.enterPowerSavingMode()
  - Sdk.this.onDestroy() -> Zello.uninitialize()
- Any class implementing the *Events* interface now needs to subscribe to *Events* updates by one of two ways:
  - By initializing *Zello* with the public static void initialize(String, Context, Events) signature.
  - By invoking the Zello.subscribeToEvents() method.
- The cancel() method has changed names to the more descriptive cancelSignIn().
- The selectContact() method has split into two because of the two methods of initialization for the Zello SDK.
  - If the SDK was initialized using the *Application Context*, use public static void selectContact(String, Tab[], Tab, Theme)
  - If the SDK was initialized using an *Activity Context*, use public static void selectContact(String, Tab[], Tab, Theme, Activity)
- Javadoc documentation is now available for the Zello SDK.