1. WEDROID TRANSITIONS
2. ZIZOTECH SOLUTIONS
3. ZIZDROID SOLUTIONS
4. ZIZTECH solutions
5. WEBDROID SOLUTION

* Launch your app at the end of the week (Thursday afternoon is usually a good time). Why so? Well, no companies would like to publish an app only 1.5 day before the week end -> too dangerous (in case there is a problem that needs a quick reaction time).
* Use proguard on your app (usually, you just have to add this line: proguard.config=proguard.cfg in the default.properties file). This will optimize, shrink and obfuscate your code, very useful for preventing from code thieves. You don't have to delete any comments, they are automatically deleted at compile time.
* Optimize your images (using [Paint.NET](http://www.getpaint.net/), [PNGCrush](http://pmt.sourceforge.net/pngcrush/) or [OptiPNG](http://optipng.sourceforge.net/)).
* Optimize your layouts for most of screen sizes. You can do this by simply changing the screen size while editing a layout in AndroidStudio or Eclipse.
* Try/catch all exceptions on the UI and display a simple toast which indicate to the user that something wrong happened. In the meantime, retrieve the error with [Crashlytics](https://try.crashlytics.com/) or something similar.
* Don't use too much .jar libraries, prefer library projects (optimize the code size) and add them using gradle.
* Prefer using vector images since it will reduce APK size and fit correctly on all devices.
* Don't use the Android preferences windows -> that's not really beautiful, even if it's in the Android guidelines, prefer making your own settings page. But if you keep Android preferences: consider adding icons and colors.
* Don't show the title of your app on the main screen (this.requestWindowFeature(Window.FEATURE\_NO\_TITLE);): good brands don't need to take so much space on a screen to be recognized (show some icon or title in the menu or somewhere that is not always visible), and consider using the fullscreen mode (this.getWindow().setFlags(WindowManager.LayoutParams.FLAG\_FULLSCREEN, WindowManager.LayoutParams.FLAG\_FULLSCREEN);) when dealing with games and very immersive content.
* Use [Google Analytics](http://www.google.com/analytics/mobile/), [Fabric Answers](https://fabric.io/kits/android/answers) or [Flurry](http://www.flurry.com/) for future analytics -> try to get as much information as possible, but don't grab anything that violates the anonymous identity of the customer. Don't forget to retrieve exceptions (errors and stack traces) that happens on the user side.
* Ask your friends to do monkey tests, learning from users usually brings many good things (such as priorities and new ideas).
* Consider publishing your app before having finished all features (most important feature only), you don't already know what your users will want or need besides your main feature .
* Add a section "More apps", or "More from developer" in your app, that's free ads.
* Add a section "Send feedback" to give the user the possibility to ask for a new feature or report some bug.
* Ask your users to translate your app by providing the strings.xml somewhere on the web like [Crowdin](https://crowdin.com/page/android-localization).
* Try your app on each Android version with the emulator -> many bugs or design issues will be detected this way. For this, you can use the provided emulator, or use [Genymotion](https://www.genymotion.com/) instead (Genymotion has a lot of very useful features).
* Think about the name of the app -> what keywords would you use to search for your app? These keywords should be the name of your app (Google will help you get discovered this way).
* Consider including keywords in the app description, but in a descriptive way (make understandable sentences using your keywords). Never add a keyword list as is in the description.
* Be the first to rate your app with 5 stars, and ask your family and friends to do the same -> this will likely influence the future users ratings.
* Consider using Google to translate your app either for the description, either for the strings.xml or both.
* Consider displaying ads in your apps and use mediation to improve your revenues [AdMob](http://www.admob.com/).
* Instead of providing a paid version, consider doing in-app billing -> users are more likely to pay in-app rather than paying for a paid version.
* Add a change log in the app -> users usually like to know what changed since the last version.
* Add a "Thanks" section for the users that helped you -> this will engage users to your product.
* Add a "If you like this app, please rate it" link (to your Google Play description) in your app -> you will get more 5 stars (usually a popup on startup, or after a feature action).
* Consider explaining your product via a "Tips" or "Instructions" section in your app.
* Save your keystore and credentials information somewhere safe. You won't be able to publish an update for your app if you lose your keystore.
* Make your icon really simple and clear. The icon is the first and also mainly the last thing that will make the user download your app.
* Unless it's not possible, prefer external installation (android:installLocation="preferExternal" in the AndroidManifest.xml).
* Read [AppAnnie](https://www.appannie.com/) tips and blog posts, it will give you hints on how to improve ASO and help your better understand your users.