# App Launch Plan – Inventory App

For this project, we've created an app for keeping track of an inventory of items, which handles multiple users and SMS alerting. Now that the app has been completed, it's important to finalize our code by testing and making sure all functionality is available and working, create a plan to deploy our app onto the app stores for public usage, and to come up with how exactly we'll monetize our app in order to keep the service up-and-running.

The app icon that will be seen on users screen and in the app store should resemble what the app is trying to do, and as a result, I think that the icon should consist of maybe a package and a database, denoting that the inventory is saved, and perhaps adding a message icon somewhere in the logo, to denote that there are messaging capabilities within the app. For the description, it should be a high-level overview of what the main purpose of the app is, as well as a few points about what exactly can be done in the app. For example, one description could be: "Keep track of your inventory counts and be alerted when counts are low! Multiple users allow for multiple separate inventories, and SMS capabilities allow users to get text message notifications when inventory items reach below a certain threshold. Download the app now to learn more!"

Our app will successfully run on any Android devices that are running API level 28 or greater. By doing so, we're targeting about 82% of devices. By doing so, we're able to make use of certain APIs, like the SMS Manager API, which allows us to send SMS text messages via the user's device to another person's number. By increasing the API level, we would be decreasing the potential number of users that would see and be able to download our app, and for very little benefit, since our app is not making use of the different components and functionality that those increased API levels would provide.

For our app, the only permissions that will ever be requested is the use of SMS via their device through our inventory app. By default, the SMS functionality will not be enabled, and the user will have to use the menu icon in the top-right part of the app to go to the "Enable SMS" screen. At that point, when hitting the "Enable SMS" button, the user will be prompted to enable the SMS permission, assuming they haven't already given it before. If they disable the functionality, even if they enabled the permission, our app will not send SMS messages. Since our app is able to completely function without any other permissions, our app will not request those additional permissions of the user.

For monetizing the app, since it's a fairly basic app, we'll likely consider going with an ad-based model, while providing a free app, so we're able to collect some sort of compensation for its usage. As the app grows, another option that could be considered would be to provide a paid version of the app, which includes more features and maybe even a customer service option for an added fee.