Week 2: Day One Assignment
Summaries.

-Human Interface Guidelines.
In order for your app to be successful, you need to make it meet the expectations for quality and functionality, and for that, there are three primary themes:
-Clarity: This means that it doesn't matter how big or small your device is, everything is legible, precise, lucid, adornments are subtle and appropriate.
-Deference: ensure that content is priority, filling up the screen, not challenge the user with its UI.
-Depth: this means that apps can be layered, impart vitality and facilitate understanding.

Someone once said, a UI is like a joke; if it has to be explained, then it's probably not good.

There are some principles to keep in mind as you imagine your app's identity;
The aesthetic integrity: It represents the app's appearance and behavior with its function. Most apps are design to prevent the user from getting distracted, games are designed to promote discovery and excitement.
Consistency: In my experience, this does not apply to apps alone, but on everything you do, as Adam Neely said: "Repetition legitimizes" (talking about music of course). In the case of the apps, implementing a consistent UI helps users know what to expect.
Direct Manipulation: Users can see results of their actions, rotating the screen, using gestures with expected results (like dragging), this engages people and facilitates understanding.
Feedback: To acknowledge actions and show results to keep people informed, this can mean having loading icons when your app is loading, to highlight something the user tapped to let him know that you listened, etc.
Metaphors: Helps the understanding of the UI when your app's objects are "metaphors" for familiar experiences, like moving a view to reveal more content, or dragging a slider, flicking a switch, etc.
User Control: The users are in control, not the apps. An app can suggest a course of action, warn the user if it must, but an app should NOT make a decision for the user. (Remember that video about how game designers protect gamers from themselves).

On a lighter note, for some reason I remembered this:
"People don't like being punished for doing something they shouldn't, they liked

being rewarded for doing something they should. WoW wanted users to take a break and stop playing, so instead of punishing them for playing too much, they rewarded them for taking a break (granting a “well rested experience boost”)”
I’m not sure how helpful that is, but it’s part of the UX design.

There are several guidelines for your app to be hosted on the App Store, but the nutshell of it is to provide a safe experience for the users, which include kids.
Do not try to cheat the system, otherwise you’re gonna get expelled from the developer program.
People want to make sure that the app they’re downloading is safe to use, not just software-wise speaking, but also morally. Avoiding content that depicts realistic violence or that promote reckless behaviors, discrimination in either political, religious, racially, among other things.
User generated content is a tricky since it may include copyrighted content, so to prevent incidental NSFW content, anonymous bullying, etc, you need a way to filter content, report abusive users, and published contact information so users may easily reach you.
Kids apps may not include items that would require an adult’s permission, which include links that take them out of the app, purchasing information, etc, unless guarded behind a parental gate. Once you launch an app designed for kids, it must keep the same rules of design even if you decide to remove it from the category in future updates.
Physical harm is not allowed, apps that can induce to spasms with flashing lights, drug usage calculators MUST come from the hospital, pharmacy, health insurance company, etc. You cannot do those out of your own wits. Apps that include DUI checkpoints are only published by government faculties.
To submit an app you have to make sure it’s complete, including all metadata, and fully functional URLs. Placeholder items should be completely replaced or removed as necessary.
Beta testing apps do not belong in the App Store. Use TestFlight instead.
Hardware capability refers to being able to run an app on all devices.
Not specifying how your app is being monetize will result on delayed reviews or even rejections. Apps with irrationally high prices are rejected. Any app that attempts to use money to unlock content should use in-app purchases. No external codes or such.

Always make your sure your code runs without warnings or errors, make your code names relevant, prioritize length over ambiguity (don’t name your variables “thingy_for_stuff”) use camelCase and not snake_case, use terms that don’t surprise experts or confuse beginners, label, comment, choose good parameter names that also serve as documentation.