Native App Studio: Design # 2

Victoria van der Mark

11 September 2015

1

The **experience** of this app, consists of (1) setting the alarm, (2) snoozing the alarm and (3) turning the alarm off

Each of the parts that together shape the experience, are called **interactions**.

These **interactions** usually consist of paths:

- 1.: (I) specifying the time, (II) choosing the wake up sound, (III) turning the alarm on.
- 2.: (I) tapping the screen
- 3.: (I) sliding the screen to the right or left

Finally, every **path** consists of a few **steps**. Since the amount of steps required for interaction 2 and 3 equals one, there is also just one path per interaction. So here, path equals steps and these will not be mentioned.

So, for interaction 1, the **steps** are:

- I.: sliding the time textview
- II.: sliding the options menu up, selecting the desired sound, sliding the options menu down
- III.: swiping the screen to the right or left

2

The app is consistent with the norms prescribed by the context in the following example:

- A smart phone application asks for a different design than, for example, a PC application. Phone apps should not be too complicated, intuitive and easily understandable. Also, all the important information (and buttons to obtain the rest) should be visible on one screen. This week's app follows this principles by having a very clean design: only basic functionality and no labyrinth of menu options.

The app is not consistent with the norms prescribed by the context in the following examples:

- The context of it being a smart phone application, prescribes traditional functionality. Traditions in design may be broken only when the benefits are clear. This app breaks with tradition, because the alarm must be set by sliding the screen to the right or left. Hiding every menu makes the app very clean, but also counter-intuitive. It takes some time before the user realizes that it doesn't matter which direction he slides the screen to. Before that epiphany, he keeps thinking: "So this means that last time I didn't slide it correctly". To maintain the clean design and still have an intuitive app, the designers could have chosen to only be able to slide the screen to the right, with a subtle indication (like an arrow).
- An app should also provide for disabled (or less abled) users. An application that uses every side of the screen also demands accessibility of each of these sides. Phones keep getting larger, and people with relatively short thumbs may encounter problems, sliding the screen and tapping the '+'-button in the upper right corner.
- Smart phones are mainly used for fast actions and should therefore respond in a fast manner. When setting the alarm, the animation is relatively long, which might become annoying when the user sees, for example, that he didn't set his alarm at the right time.
- As stated before, a smart phone application should be clean and easily understandable. However, there is a '+'-button in the upper right corner, that doesn't seem to have any functionality. This can be very confusing for the user; he might think that he is missing out on something or that he doesn't understand the application.

3

An affordance is a part of the design of an object, that allows the user to interact with it (and thus with our surroundings).
Examples of affordances in Rise are:

- A swipeable main activity, which affords swiping for setting the alarm
- The settings list is slideable, which affords sliding it up to see the possible settings.
- The time the alarm is set is moveable by means of touch, which affords sliding it up and down for specifying the time of the alarm.

- When the alarm rings, the screen is tappable, which affords tapping on the current time to snooze for 5 minutes.

The affordance op tapping the screen could be improved, by being able to not only tap the screen, but slide it down for specifying the duration of the snooze.

4

The designers may have used the following constraints to shape the app's design:

1. People have personal preferences when it comes to being woken up. The app should therefore serve all these desires by offering (for example) different waking-up sounds.
2. Sleepy people are often a bit confused and not fully capable of answering complex questions. The options a user can choose from when the alarm sounds, should be very intuitive and simple.
3. When sleepy, people tend to be more negative. Therefore, the focus must lay on satisfying woken up people, so they keep using the application.
4. Everyone needs an alarm, which means that there is a very large group of potential buyers. Therefore, the app should be satisfying for beginning, intermediate and expert users. For this purpose, the app should be intuitive, but also pretty and sophisticated.
5. People mostly look at the app when they have just woken up, so the intensity of the colours should be very low.

5

A concept tries to capture an (often abstract) phenomenon. It does not stand for a specific object, but must be further elaborated to obtain a meaning.

6

The following list consists of ten important concepts mentioned in the book, with a short explanation. They are rated on a scale from 1 to 10, on how easy it would be to use the concept in building an app.

- mission statement: defining what you want to make in advance. 9
- behaviour: a product's response to input. 8.5
- controls: an element triggering behaviour through means of an interaction. 8

- layout: the way elements are arranged. 7
- interaction: the paths taken when interacting with a product, from starting it to shutting it down. 6
- mapping: connection between related controls and their corresponding behaviours. 8
- good fit: responding to normative context. 7
- responsive web design: dynamically adapting the layout to your devices width/height. 7
- affordance: anything that allows and encourages an action to take place. 6
- pattern: a repeatable solution to a common design problem, taking into account the implications of the problem. 6