The idea is to design an app that looks like a game.
So the app offers the user a variety of colors, and the user have to pick one as the target color.
Then the app asks the user which one of the three primary colors (r, g, and b) looks closer to the target color they've picked. The answer would be the start color. Then the game begins.
The user gives instructions, in the form of text, to the app to change the start color step by step in a way that they finally reach the target color.
There's also a scoring system for the shortest path to the target color or to get close enough to it.

The reason I decided to work on this idea is that 'Color perception' is a very interesting topic, there are many studies focused on analyzing how people perceive colors and how demographic characteristics might affect their perception.
And the beauty of these studies is that the colors are in fact measurable phenomena in a very objective way, right? The wave frequency.
It's like having a firm ground to stand on and measure how, say my gender or race, changes my perception or even just expression of the color yellow or a fixed measurable difference between say blue and green, in a different way from someone with a different gender and race.

Anyway, the learning scenario here is more about a successful communication, between the user and the machine. The user needs, first, to think clearly, and then to express themselves clearly, as well. And by repeating the game they might also learn a few rules about color combinations. Plus, some eccentric color names that were collected through xkcd website survey by some hundreds of thousands of participants for naming monitor rgb colors.

But to be honest, the learning part that I'm more interested in, is on the machine side. That is the data we could collect from users playing this app. The way they express colors, could be a precious resource for teaching the machine some sort of color common sense, I would say. For example, a user might say something like: 'the target color is more lively.' Then the app would be like: 'I don't get it, what does lively mean' so demanding the user to express 'lively' in more basic terms. But it will add 'lively' to its 'adjective' pool, with a proper corresponding action/change.

For the technology let's have a look at code

And for the interface, I really like to work with the MIT App Inventor, which is a very easy, cool, and fun way of making apps. If the connection helps, I can show you how cute it is.
But if I feel like I need to use some NLP libraries, that I, apparently, cannot import to MAI, I would leave the nice interface for the sake of a fluent user experience.