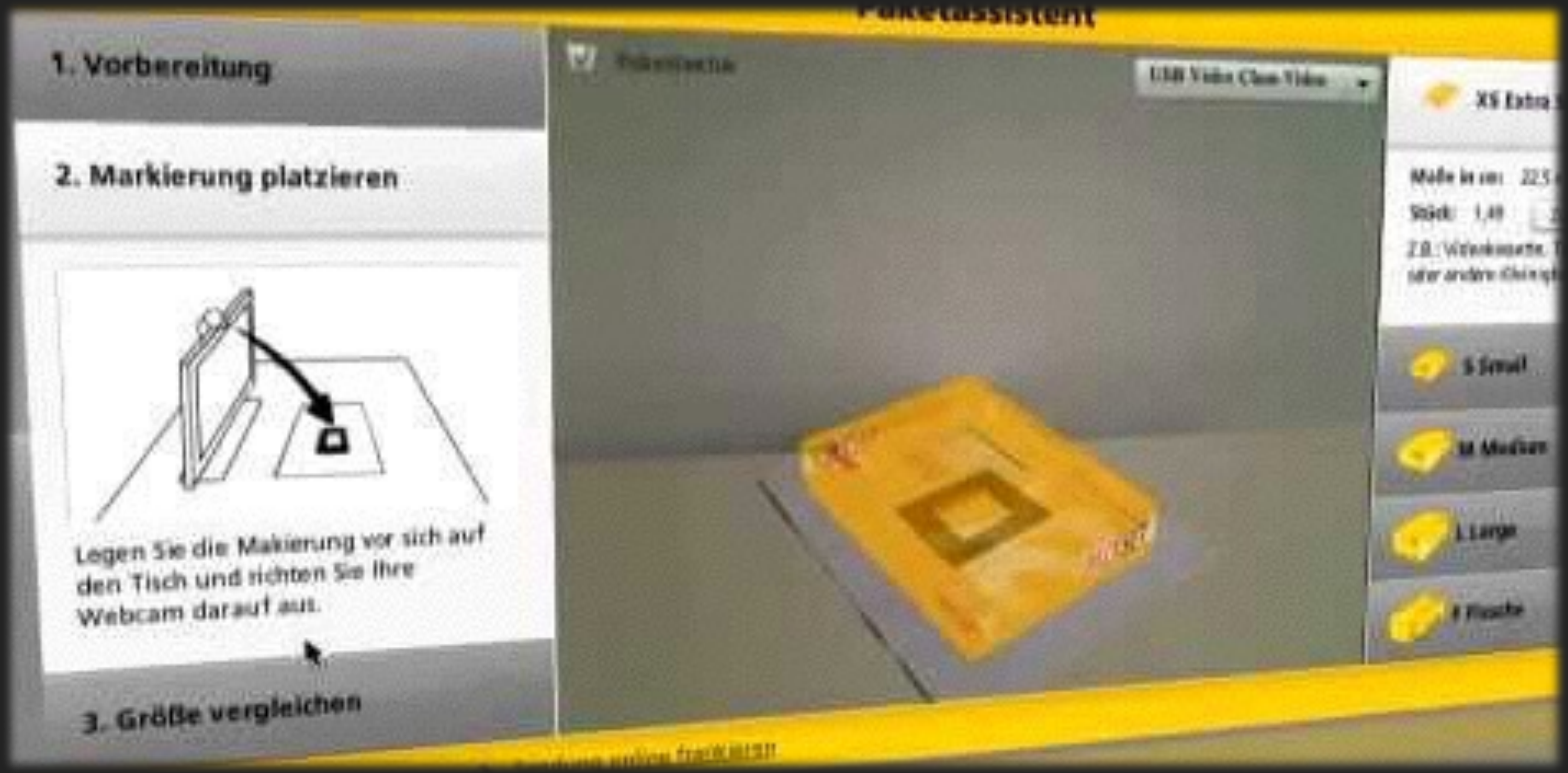
ANGELO CAMMALLERI

Junior iOS Developer
Interactive Pioneers

DEVELOPING AN ARKIT APP

CREATION OF AN IDEA

- An old idea from 2012
- Finding the right package without actually measuring
- ARKit allows to streamline the usage as no marker is needed



1. Vorbereitung

2. Markierung platzieren

Legen Sie die Makierung vor sich auf den Tisch und richten Sie Ihre Webcam darauf aus.

3. Größe vergleichen

Video

STARTING FROM ZERO? HELL NO.

- With ARKit Apple also released the Placing Objects Demo to developers
- It is a great boilerplate for starting a small AR project
- Free and open: MIT license
- For the hacker types - It is still a good inspiration
- Download (GitHub)

WHAT DOES PLACING OBJECTS OFFER?

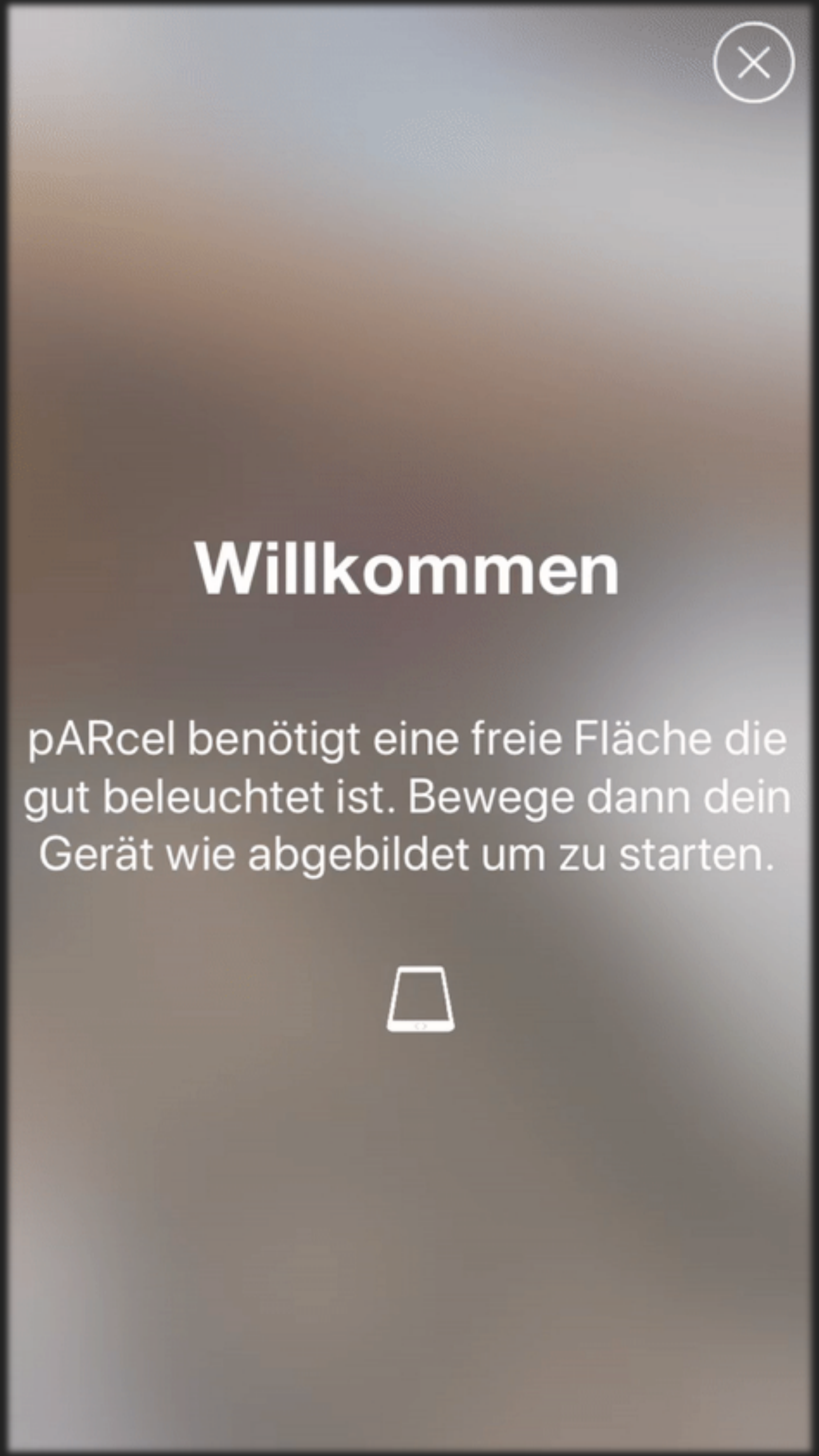
- Developers with zero SceneKit knowledge see actual implementation
- Code with ARKit delegate usage
- All possible states of the ARKit tracking are handled and displayed in a basic matter
- You get a focus square for free
- Head start for your app vision

CALIBRATION PROBLEM

- For our use case a plane is required
- How can we make the initial plane detection easy to understand for users?
- What is the quickest way to calibrate?

CALIBRATION SOLUTION

- Upon launch of the app we show a help screen
- It demonstrates how to move your device for best detection results
- Detection is all ready running during display
- If a plane is found the screen is hidden immediately



Willkommen

pARcel benötigt eine freie Fläche die gut beleuchtet ist. Bewege dann dein Gerät wie abgebildet um zu starten.

PLANE DETECTION PROBLEM

- It is running all the time by default
- Placing a physical object can result in a new plane on its top
- Repositions SceneKit objects to that new plane
- This breaks our use case

PLANE DETECTION SOLUTION

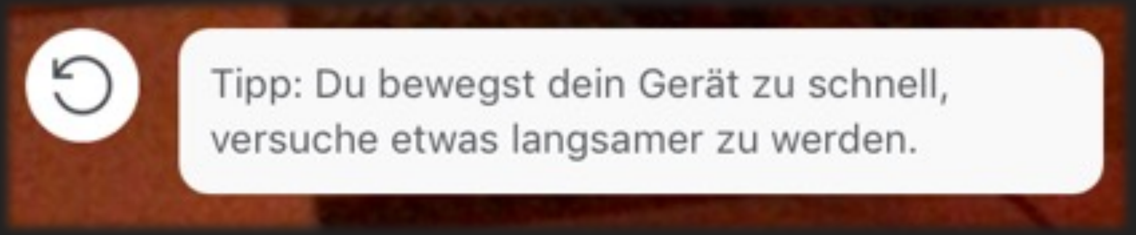
- When a plane is found we fade in our package
- A few seconds after that we deactivate the plane detection
- When the user is ready to place an object the detection is all ready turned off

TRACKING STATES PROBLEM

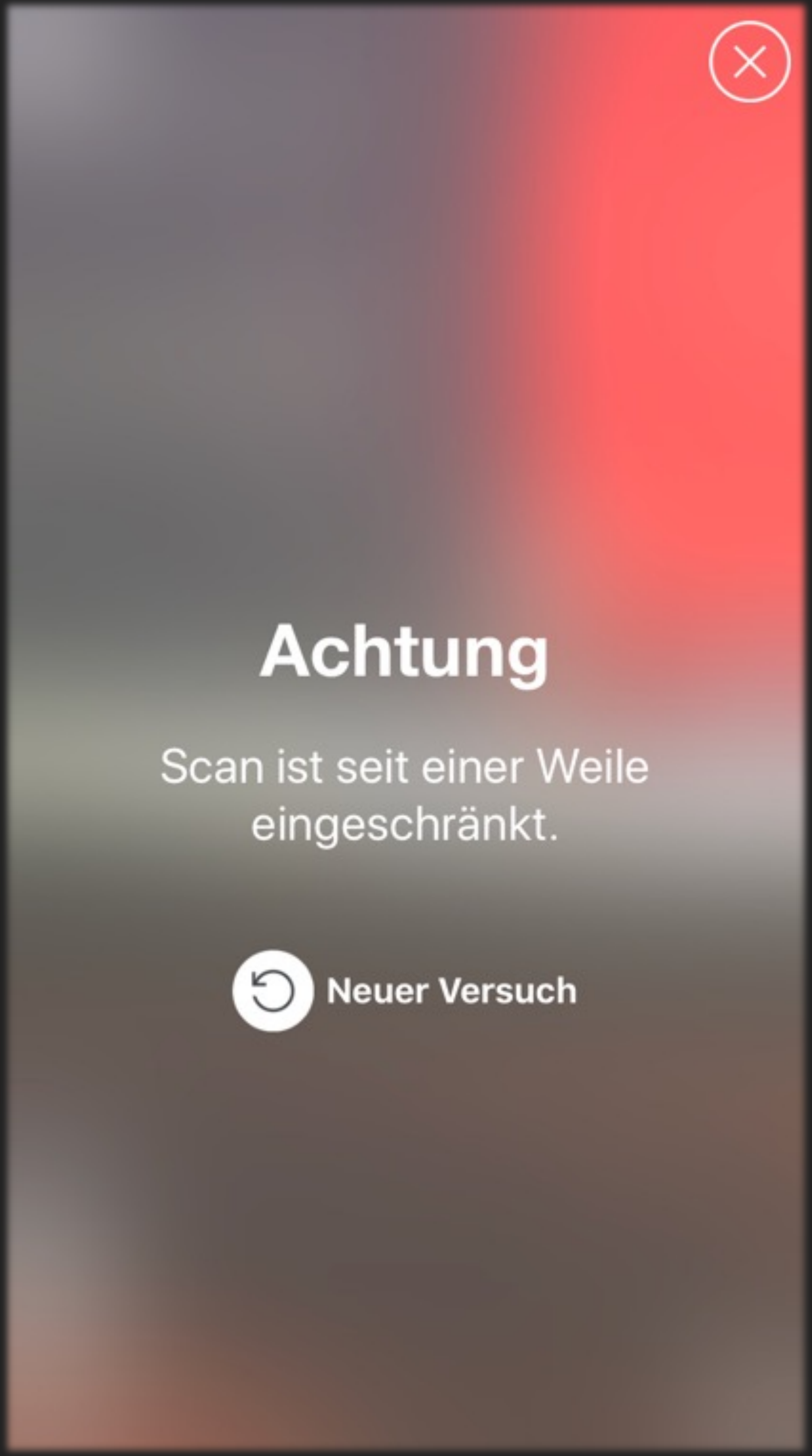
- Making different tracking states understandable for users
- You have to handle warnings and errors
- Sometimes you want to give the user a hint

TRACKING STATES SOLUTION

- We divided the tracking states into 2 categories and added one
- Hint, Warning and Error
- Displaying most hints and all warnings at the top without disrupting
- Errors and the welcome screen as overlay
- Strong haptic feedback is played when an error occurs
- Light feedback for hint overlays like the welcome screen



Tipp: Du bewegst dein Gerät zu schnell, versuche etwas langsamer zu werden.



Achtung

Scan ist seit einer Weile eingeschränkt.

INTERACTION PROBLEM

- We want our users to interact with 3D objects like they would expect
- Pinch and pan resizes an object by default but we have to be accurate with object size
- So we need a different action performed in that case

INTERACTION SOLUTION

- When the user pinches or pans a different package is selected
- In comparison to the default situation of resizing an object here it is replaced
- So we had to save the exact position and rotation of the object and apply it to the new one
- To raise awareness that a switch succeeded we added haptic feedback

DEMO

THE RELEASE VERSION

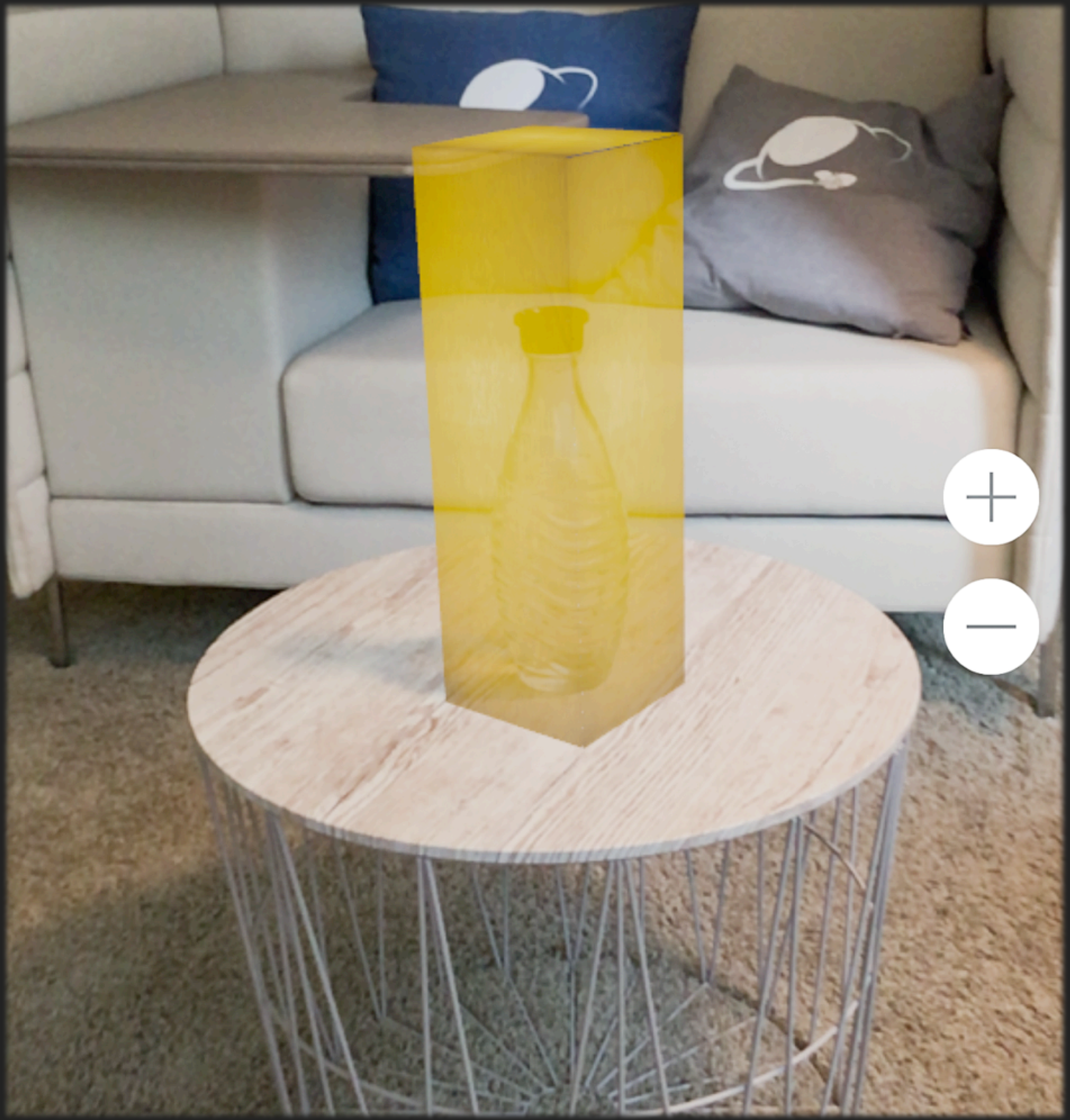
- Prototype
- App Website
- Download

WHERE TO GO FROM HERE?

- Implementing an own focus square
    - Custom style
    - Custom reaction to situations (Plane detected)
- Better guiding of users during calibration phase
- Extended selection of packages
- Postage calculation depending on size selection
- Provider selection for postage

THE IMPACT OF AR

- AR as technology is not new
- Broad consumer device support is
- "It is likely that in 6 months time, close to a quarter of a billion devices will be able to run ARKit powered apps." - Owen van Dijk
- Game Engines are for games right? Nope.
- Find talented 3D Artists now!



parcel

THANKS FOR JOINING ME

QUESTIONS AND FEEDBACK ARE GREATLY APPRECIATED.