



## POC Arduino mit Scanner und Verbindung zur Firebase

	<b>Aktivierung eines Scanners</b>
Beschreibung	Der ESP32 wird an eine Stromquelle angeschlossen
Exit Kriterien	Der Scanner zeigt dem Benutzer durch Aufleuchten des weißen Lichtes, dass er startbereit ist
Fail Kriterien	Der Scanner leuchtet nicht auf und der Benutzer bekommt kein Feedback
Fallbacks	Die Verbindung von ESP und Scanner muss überprüft und gegebenenfalls korrigiert werden. Stromversorgung überprüfen.
Durchführung	<ol style="list-style-type: none"> <li>1) Der ESP32 wurde an eine passende Stromquelle angeschlossen. Der Scanner leuchtet rot und weiß. Der Scanner war einsatzbereit.</li> <li>2) Der ESP32 wurde an eine passende Stromquelle angeschlossen. Es erleuchtete kein Licht. Anschließend wurden die Stromversorgung und die Verbindung zwischen ESP und Scanner überprüft. Das „GROUND“-Kabel zwischen ESP und Scanner hatte sich gelöst und dies wurde korrigiert. Nach eines neuen Versuchs wurde das Problem gelöst und der Scanner leuchtete rot und weiß auf.</li> </ol>

	<b>Korrekte Platzierung von Scanner und Barcode</b>
Beschreibung	Der Barcode wird <b>in einem 90°-Winkel (gerade) vor</b> den Scanner gehalten
Exit Kriterien	Der Produktcode wird korrekt erfasst
Fail Kriterien	Der Produktcode kann nicht erfasst werden und der Barcode muss neu vor den Scanner ausgerichtet werden
Fallbacks	Andere Platzierung des Scanners oder Barcode, sodass letzterer sich im Scan Feld (weißes Licht) befindet.
Durchführung	<ol style="list-style-type: none"> <li>1) Der Barcode wurde in 90°-Winkel vor den Scanner gehalten. Danach wurde der Barcode in mehreren Abständen von je 1cm vor den Scanner wegbewegt (Beginnend bei 0cm). Der Scanner war fähig den Barcode in einem Abstandsbereich von 5-20cm zu lesen. Der Produktcode wurde erfasst und der Nutzer erhielt eine Output Indication zur Rückmeldung.</li> <li>2) Der Barcode wurde in verschiedenen Winkeln (&lt;90°, &gt;90°) vor den Scanner gehalten. Die Barcodes konnten nicht eingelesen werden.</li> <li>3) Der Barcode wurde in einem 90°-Winkel, mit einem Abstand von 10cm vor den Scanner gehalten. Zwischen Scanner und Barcode befand sich ein eingefärbtes Glas. Der Barcode konnte nicht gelesen werden.</li> <li>4) Der Barcode wurde in einem Abstand von 0cm vor den Scanner gehalten. Der Barcode konnte nicht gelesen werden. Das gleiche Ergebnis wurde in den Versuchen 1-4cm und &gt;20cm erzielt.</li> </ol>



	<b>Rückmeldung des Scanners an den Benutzer</b>
Beschreibung	Der bereits im Scanner verbaute Buzzer gibt dem Benutzer bei erfolgreichem Scan eine Rückmeldung
Exit Kriterien	Der Barcode wird erfasst und der Buzzer gibt eine Rückmeldung
Fail Kriterien	Der Barcode wird erfasst, der Benutzer bekommt jedoch keine Rückmeldung des Buzzers
Fallbacks	Scannen des Barcodes für OUPUT INDICATION (Buzzer), um die gewünschte Rückmeldung zu erhalten
Durchführung	<ol style="list-style-type: none"> <li>1) Der Barcode wurde in korrektem Abstand und Winkel vor dem Scanner platziert. Der Scanner gab ein haptisches Feedback (Buzzer) zurück.</li> <li>2) Der Barcode wurde in abweichendem Abstand und Winkel vor dem Scanner platziert. Der Scanner gab kein haptisches Feedback (Buzzer) zurück</li> </ol>

	<b>Aufleuchten einer LED</b>
Beschreibung	Bei <b>erfolgreichem</b> Abscannen des Barcodes erhält der Benutzer <b>immer</b> eine visuelle Rückmeldung in Form einer aufleuchtenden farbigen LED (grün <b>oder</b> rot)
Exit Kriterien	Der Barcode wird erfasst und eine LED beginnt zu leuchten
Fail Kriterien	Der Barcode wird erfasst und keine LED beginnt zu leuchten
Fallbacks	Überprüfung der Komponenten LEDs, wenn nötig sind diese auszutauschen
Durchführung	<ol style="list-style-type: none"> <li>1) Der Barcode wurde (siehe Korrekte Platzierung von Scanner und Barcode) korrekt vor den Scanner gehalten. Eine der LEDs leuchtet auf.</li> <li>2) Der Barcode wurde erfolgreich abgescannt. Es leuchtet keine LED auf. Nach Überprüfung der Verbindung und erneutem Anschluss der LEDs zum ESP32 wurde der Vorgang wiederholt. Der Barcode wurde erfolgreich abgescannt. Eine der LEDs leuchtet auf.</li> </ol>



	<b>Benutzer speichert Daten in Realtime Database (RD)</b>
Beschreibung	Der Switch befindet sich auf Position „I“. Der Benutzer kann verschiedene Barcodes speichern. Die grüne LED leuchtet auf, wenn der Barcode erfolgreich in der RD gespeichert wurde. Es wird ein neuer Barcode mit der Anzahl 1 gespeichert. Falls dieser Barcode bereits in der Database vorhanden ist, wird die Anzahl um 1 inkrementiert.
Exit Kriterien	Der eingelesene Barcode wird in der Datenbank gespeichert
Fail Kriterien	Die RD kann nicht erreicht werden, die rote LED leuchtet auf
Fallbacks	Überprüfung der Internetverbindung des ESPs, ggfs. muss der Standort des ESP32 so verändert werden, dass eine stabile Internetverbindung gewährleistet ist.
Durchführung	<ol style="list-style-type: none"> <li>1) Ein Barcode wurde eingescannt und die grüne LED leuchtete auf. Nach Überprüfen der Firebase Konsole wurde bestätigt, dass der Barcode erfolgreich abgespeichert wurde.</li> <li>2) Ein Barcode wurde ohne aktive Datenverbindung abgescantt. Die rote LED leuchtet auf. Nach Überprüfung der Firebase Realtime Database wurde festgestellt, dass der Barcode nicht abgespeichert wurde</li> </ol>

	<b>Benutzer verändert die Anzahl eines Barcodes</b>
Beschreibung	Der Switch befindet sich auf Position „O“. Der Benutzer kann durch Einscannen eines Barcodes die Anzahl eines bereits in der Datenbank vorhandenen Barcodes um 1 dekrementieren. Die geringste Anzahl eines Barcodes in der Datenbank beträgt 0
Exit Kriterien	Die Anzahl eines Barcodes wird um 1 dekrementiert
Fail Kriterien	Die RD kann nicht erreicht werden, die rote LED leuchtet auf und die Anzahl bleibt unverändert
Fallbacks	Überprüfung der Internetverbindungen des ESPs. Überprüfen der Inhalte der Firebase
Durchführung	<ol style="list-style-type: none"> <li>1) Ein Barcode wurde eingescannt und die grüne LED leuchtete auf. Nach Überprüfen der Firebase Konsole wurde bestätigt, dass die Anzahl des Barcode erfolgreich um 1 dekrementiert wurde.</li> <li>2) Ein Barcode wurde ohne aktive Datenverbindung abgescantt. Die rote LED leuchtet auf. Nach Überprüfung der Firebase Realtime Database wurde festgestellt, dass die Anzahl des Barcodes nicht dekrementiert wurde.</li> </ol>



	<b>Einbindung der API</b>
Beschreibung	Ein request wird an die OpenFoodFacts-API geschickt
Exit Kriterien	Die Datenbank gibt den Statuscode 200 zurück
Fail Kriterien	Der API-Anbindung ist nicht korrekt, es gibt einen Statuscode 4XX zurück
Fallbacks	Die URL der OpenFoodFacts muss überprüft und ggfs. korrigiert werden
Durchführung	<ol style="list-style-type: none"><li>1) Der request war erfolgreich und erhält eine response (statuscode 200)</li><li>2) Der request war nicht erfolgreich (statuscode 4xx)</li></ol>

	<b>Darstellung der Datenbank auf einer Webseite</b>
Beschreibung	Die von der API umgewandelten Daten werden dem Benutzer auf einer eingebundenen Webseite zur Verfügung gestellt
Exit Kriterien	Die korrekt ausgelesenen API-Datensätze zu den passenden Barcodes werden auf der Webseite angezeigt. (Statuscode 200)
Fail Kriterien	Die Webseite gibt einen Fehler beim Lesen der HTML-File zurück (interner Serverfehler, Statuscode 500)
Fallbacks	Korrigieren des HTML-File Pfads
Durchführung	<ol style="list-style-type: none"><li>1) Die Website wird mit korrektem Inhalt angezeigt</li><li>2) Die Website wird nicht mit korrektem Inhalt angezeigt, die Konsole erzeugt einen Error (Fehler beim Lesen der HTML-File, die Webseite zeigt den Schriftzug „interner Serverfehler“)</li></ol>