**ID.me General Questions:**

1. What does an ideal in-store customer experience look like for you?
2. Are there standardized ways that you offer discounts in-store today?
3. You currently only have about four branded HOKA stores — are there plans to expand that footprint?
   1. **US:** Orlando, Florida; Costa Mesa, California; Venice, California; Honolulu, Hawaii
   2. ***Nina:*** *we have more than four stores? Flatiron NY, Flagship store NY, Boston MA, Orlando FL, Miami, Chicago IL, Las Vegas NV, Studio City CA, Melrose CA, South Coast Plaza CA, Toronto, ON Canada*
4. Do you see in-store verification as only for owned stores, or also for your retail partners who sell HOKA (e.g., Fleet Feet, Dick’s Sporting Goods, specialty shops)?
5. If expansion to retail partners is on the horizon, do you envision a standardized process or will each partner’s POS need to be addressed individually?
6. Are there particular retail partners where extending this verification program would be most strategic?
7. What would success look like for the in-store program?
8. Any brands that Hoka considers to be their north star or ideal state?

**HOKA Questions + Answers:**

* **Q:** Do you see meaningful trade-offs between Option A and Option B in terms of customer experience, security, or operational overhead?
  + **A: Option A** does require some initial effort but does provide the least amount of work after. It also minimizes fraud compared to Option B, since the user verifies on the spot (instead of receiving a coupon). If you can get that JSON payload into the POS system “quickly”, when the shopper verifies, the cashier can simply verify the user's email address and apply the coupon.
    - Good user experience
    - High security
    - Low operational overhead but initial heavy set up
  + **Option B** is quicker to set up and get up and running, however, there is a higher level of “code sharing” since once the user verifies and [ID.me](http://id.me) produces a coupon, that coupon can be passed to a non-community member to utilize at checkout
    - Decent user experience
    - Mid security
    - Mid operational overhead but quick initial set up
* **Q:** Are there any quick-start or pilot approaches other retailers have used to test this before committing to a full integration?
  + **A:** Go with the hosted landing page (Option B)
* **Q:** From a timeline perspective, what would be a realistic range for enabling each option if we started exploring now?
  + **A: For Option A, we will want to chat with the technical team to see if they’re able to get the JSON payload data to the POS system + how quickly the API calls happen from CRM to POS to ensure a good user experience for in-store verifiers.**
* **For Option B, we can get this up and running much quicker, however, your team will have to manage codes and upload them when they run low. Recommended for quickly getting up and running.**

**Questions on Option A:**

* **Q:** From your perspective, what would be required for HOKA to retrieve the JSON payload (step 6) and pass that into our POS environment? Would this typically be a custom integration, or do you have out-of-the-box patterns?
  + **A:** As the JSON payload is already being passed to your CRM, the question our team will have is if HOKA can pass that information to your Point of Sale (Predict Springs?) to ingest and display
  + Also, if there is a delay in getting the JSON payload to your POS
  + Use cases:
    - **Shopper in-store** verifies using a QR or RFID code - how long will it take for the verification to get to the POS
    - **At home shopper, looking to go to the store** verifies at home and plans to go into the store after work
* **Q:** Do you have existing retail partners who’ve integrated JSON payloads into their POS systems? If so, what approaches have worked best?
  + **A:** Our automotive partners will allow verification online and the information is synced with their dealership systems. However, there is a delay in the sync, so it isn’t instant, ex: Hyundai
* **Q:** What are the expected technical dependencies on our side?  e.g., APIs, middleware, or POS modifications
  + **A:** If there is a two-way communication with your POS and CRM (where your payload currently resides), then you can utilize the email they share to activate the discount
* **Q:** Roughly what level of effort would you anticipate for a retailer to enable this end-to-end?
  + **A:** Mid to high depending on how the technical team can manage the JSON payload. Per the documentation of Predictive Spring, you potentially could use API to pass the data payload over and activate the discount

**Questions on Option B (Overall easier and quicker to get up and running):**

* **Q:** For step 6 in the hosted flow, can you walk us through in detail how the barcode or generic QR code vending works in practice? For example, is it dynamically generated per shopper, reusable, or tied to their ID.me wallet?
  + **A:** HOKA provides a CSV with numeric codes to [ID.me](http://id.me) and once the shopper verifies, [ID.me](http://id.me) will dispense that code in barcode form. The shopper will present the code either printed or on their phone for the cashier to scan to apply the discount
  + Confirm if Predictive Spring can generate codes and scan bar codes within store to apply the discount
* **Q:** Do you support multiple formats, e.g., scannable barcodes, alphanumeric codes, QR codes? How do retailers typically configure which one works best for their POS?”
  + **A:** Numeric codes can be shared with [ID.me](http://id.me)
  + [ID.me](http://id.me) use those codes to dispense barcode versions
* **Q:** What dependencies or setup are required on our side to accept these barcodes/QR codes at the register?
  + **A:** Your POS should be able to scan barcodes and automatically incorporate the discount
  + Codes can be one time use per in-store visit
* **Q:** What is the estimated effort to enable the hosted landing page flow compared to the integrated option?
  + **A:** A fraction of the effort upfront, however, HOKA will have to log into [ID.me](http://id.me) to upload the promo codes when they run low (we have automated notifications)