Create a web front-end leveraging a list of in stock items that they have produced for you as JSON, and use a client s ide search feature.
Have the kiosk call an API to get a product's Amazon rating or review by id.
Allow staff to login (using Cognito) at the kiosk, so they can further request a report.
You will have a background process (AWS Step Functions) do all the various steps, and build out an HTML page th at will get uploaded it to a private area on Amazon S3.
Have your background process create a pre-signed URL for that report in S3 and send that URL (via Amazon SNS) t o their cell phone that is gathered from Amazon Cognito.

LAB 1 - Get the front end up, by uploading a simple front-end where a user can search for an item.
LAB 2 - Get the plumbing in place, by creating three placeholder API Mocks. You will have the website hit up 3 dis tinct API endpoints that just pipe back the same dummy data for reviews and ratings and create report regardless of what product ids your front-end sends.
LAB 3 - Set up the authentication for staff using Cognito user pools. Wire that into the create report API endpoint an d have it reject non authorized requests.
LAB 4 - Replace the mock AWS Lambda Functions with functions that use S3 SelectObject to return real data to the 2 GET endpoints: get_reviews and get_ratings.
Lab 5 - Have the create_report API kick off a step function background process. Have the step function call various Lambda functions that do various things. Such as sentiment analysis on a review. Look for key phrases to help with tagging in the report, and create the HTML report and pre-sign it. Finally sending that URL to the logged in user's ce llphone.
Lab 6 - Add metrics to instrument how it is performing (AWS X-Ray). Find ways to improve one of your Lambda f unctions with context reuse, and cache the response.