🔒 **monsooncommerce** / **monsoon-mvp** `Private`

Month 02 Milestone #72

Edit   New issue

⊙ **Open**   **jseiffert** opened this issue on Aug 13 · 4 comments

---

jseiffert commented on Aug 13 • edited ▾

First, let's recap the first milestone of the MVP.

For the second milestone, the user will be able to list the products they added on Amazon.com.

- The user will be able to add SKU metadata as part of the "Add a product" flow.
  - The SKU metadata includes the:
    - SKU (required)
    - Condition (required)
    - Quantity (required)
    - Price (required)
    - Cost of goods sold (optional)
    - Cost of sale (optional)
- The "Add a product" flow will validate the file to see if it can be processed.
- All products will be listed on Amazon.com. This will require the SKU metadata, ASIN, a hard-coded shipping template ID. Any other required fields by Amazon will also be hard-coded.
- When the user views their products, they'll be able to see if a product is active (listed) on Amazon.com or is inactive (A product that is not listed may be because the app hasn't listed it yet or Amazon returned an error).

We're going to implement the following UX as part of this milestone:

- The layout will be implemented (nav bar, title bar, etc)
- The product details panel will be implemented
- Pagination will be implemented
- We may also implement the active and inactive saved search. This will depend on the scope of implemented the other features.

The goal for having the second milestone be feature complete is September 31, 2018.

🏷 **mandrews** added this to the **month_02** milestone on Aug 13

🏷 **mandrews** added the `enhancement` label on Aug 13

🔖 This was referenced on Aug 20

Monsoon React Demo #96   `⑂ Closed`

"Add a Product" Upload File Format #111   `⊘ Closed`

---

chris-ramon commented on Aug 27

Cool, thanks for filing the Milestone #2 **@jseiffert!**

I've started to dig into it and next sharing my thoughts on the possible actionable steps:

Listing on Amazon

1. Convert the current products format to amazon submit feed format:

Assignees   ⚙

No one—assign yourself

Labels   ⚙

`enhancement`

Projects   ⚙

None yet

Milestone   ⚙

month_02

Notifications

3 participants

- Observations to have in mind: structure validation, scalars validation, optional/required validation and file size.

2. Submit the products to Amazon, using the current lib of choice: `amazonMws.feeds.submit(...)` :

  - Observations to have in mind are: throttling, file size and content-type, details here.

👍 2	

🔖   👤 **mandrews** referenced this issue on Aug 28

Product Listing as Background Process #114   🚫 Closed

mandrews commented on Aug 28 • edited ▾

@chris-ramon I definitely think those are good actionable next steps 👍 . I was also thinking that we might want to consider how to perform Product Listing in the background as we might have to handle a potentially large upload where the MWS calls could timeout and/or need to be retried. I tried write up a possible background process as an RFC:
#114

🔖   👤 **jseiffert** referenced this issue 26 days ago

Month 03 Milestone #185   ⓘ Open

mandrews commented 15 days ago • edited by chris-ramon ▾

Recap of our critical path tasks:

- ☑ 0. fix AWS Amplify version issue
- ☑ 1. finish up Milestone 02 styling
- ☑ 1. finish implementation of `amazon_product_lookup`
- ☑ 1. merge MWS product listing branch
- ☑ 2. connect MWS product listing to AppSync
  - ensure tenant id, cognito id can be passed to library
  - can skip persistence and background processing as first step
- ☑ 3. migrations for product listing table
- ☐ 4. connect MWS product listing to persistence
- ☐ 5. background polling / update of MWS product listing
- ☑ 5. decide on schema for product table - this is the table that is exposed in Product Listing
- ☑ 6. migration for product table
- ☑ 7. add getProducts to GraphQL schema (#213)
- ☑ 7. add getProducts to AppSync (#213)
- ☑ 8. connect UI to getProducts (edited)

mandrews commented 7 days ago • edited by chris-ramon ▾

Updated Recap of Critical Path Tasks (number denotes parallelism):

- ☑ 0. Integrate monsoon-appsync's `products` GraphQL query to `/products` page. #235
- ☐ 0. Initial productsLookup with submitMarketplaceFeed integration #236
- ☑ 0. Initial Initial FeedProcessing integration #237
- ☑ 0. monsoon-ui: null checks missing on `/products` page #232
- ☐ 1. Persist result of submitMarketplaceFeed #238
- ☐ 2. Persist result of FeedProcessing #239

Nice to Have Tasks:

- ☑ 0. monsoon-ui: side menu links only working on clicking the link text #233