

- \* Front-End
- \* Sanity
- \* Shipment API

Using Hackathon 1 Template:

\* Front-end:

- ✓ - Responsive for mobile and desktop
- ✓ - User Friendly Interface
- ✓ (70%) - Proper Pages for each Section  
(checkout, Order conf. left only)

\* Sanity:

- Products Page: Using Sanity, the data on product page will be displayed instead of static data
- Product Detail Page: The data of each product would be displayed dynamically using catch all segment route.

• Checkout: The user would

• Cart Page: Cart would be managed by taking - Product Name, Quantity, Price (Ref) and storing the info in the local data. Add to Cart would also store in local data.

• Checkout: When user clicks on checkout in Cart Page, a random string would be generated as Order ID. Local data would be transferred to Sanity.

This gives 2 OPTIONS:

Successful Payment

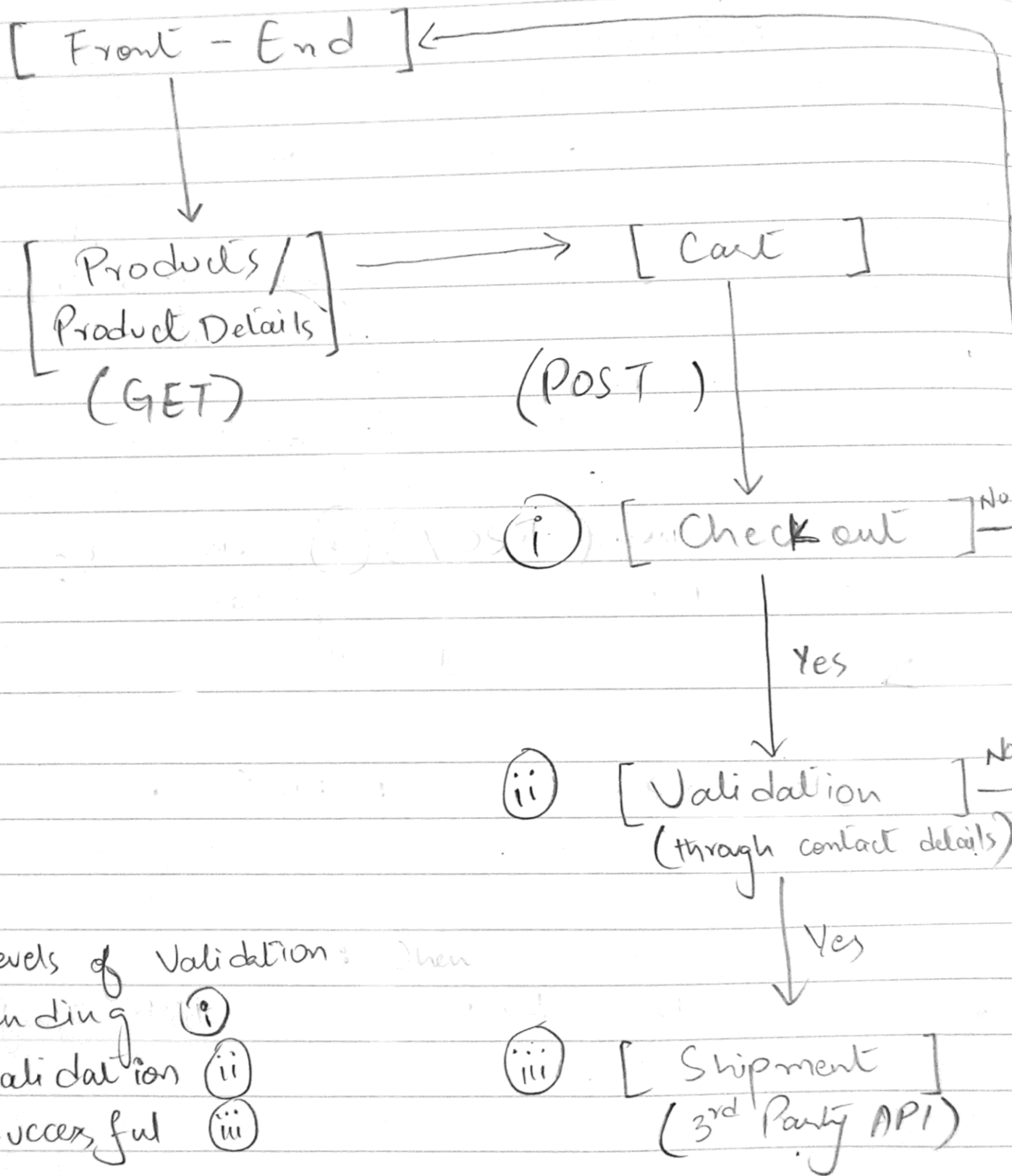
If checkout successful, order status is updated on Order ID.

\* Also, using the user's details the ORDER ID will be mailed to him along with Tracking details

Cancellation

Session expires and user is sent back to homepage. Data is deleted from Sanity.

\* Human validation done by contacting - Preventing Bot attacks



\* 3 levels of Validation: Then

- Pending (i)
- Validation (ii)
- Successful (iii)

\* Online Payment is not included, however local Payment Systems like Easypaisa and Jazz cash can be integrated.

+ At the moment, Payment will be COD for simplicity.

## \* Work Flows :

- ① User visits our Marketplace.
- ② He chooses Products according to his needs.
- ③ The Products are stored in local database.
- ④ When user visits the Cart Page, the items from local storage gets logged there.
- ⑤ Item Quantity, Name and Price are shown.
- ⑥ User Proceeds to checkout, here the details from local storage are POST to Sanity.
- ⑦ Order ID is generated.
- ⑧ User checks out and Order details are mailed to him with Tracking details. Validation through Call is done.

⑨ IF User Cancels checkout,  
the data from Sanity is ~~DELETE~~

## API ENDPOINTS:

① ENDPOINT: /products  
Method: GET

② ENDPOINT: /cart  
Method: POST

③ • POST when user 'checkouts'

③ ENDPOINT: /checkout  
Method: POST / DELETE

• POST to update status

• DELETE when cancelled