**Problem Statement:**

**Decentralized E-commerce website**

Create a decentralized E-commerce website where people can make payments for products in cryptocurrencies (Neo/Ether/BTC, etc.).

**The following design is suggested for the application-**

* Design a home page that contains the details of the products. Users should have an option to search for products according to product name/product-category/etc.
* Every product should have details like product image, price, description, and a number of items available.
* Create an admin page that can add new products to the website with their complete details.
* Whenever a new product is added to the website, its metadata should be stored on the blockchain.
* Create a user login page that enables any user to log in to the website and buy products in cryptocurrency. All user transactions should be stored in the blockchain.
* The money will be temporarily stored in the wallet address of the E-commerce website smart contract whenever a sale is made, which will then be sent to the wallet address of the product seller from the E-commerce website smart contract wallet.

**What to Submit**

* Use any blockchain of your choice to store all cryptocurrency transaction details. Data excluding the transaction details can be stored on any centralized database of your choice.-------🡪 **Done**
* Integrate Meta mask for wallet transactions.-------🡪 **Done**
* Create the complete working User Interface for the website.--------**🡪 Done**
* You can use Hardhat/Truffle for getting test cryptocurrencies.--------**🡪 Done**
* Provide a URL to your working prototype.
* Provide a URL to your GitHub code repository. The repository must be public
* Include a video (3-5 minutes) that demonstrates your project in action (hosted on YouTube, LinkedIn, or Facebook Video and made public). The video must include an explanation.

**Solution:**

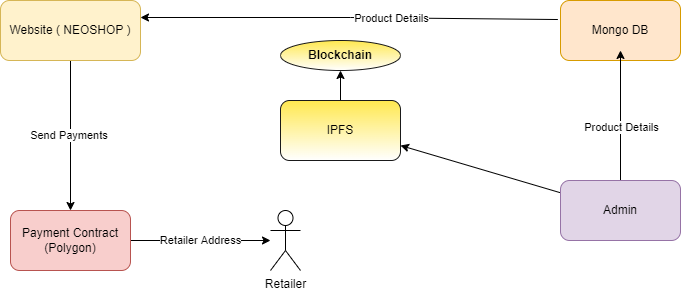
**Introduction:**

Humans are the only species who live in groups in a more collaborative structure exchanging different goods and services. Trade is core to human functionality as civil society and underlying trade that keeps maximum participants aligned to some rules.

**Web3** revolutionalizes every industry as tokenization of assets in both digital and physical realms will enhance and ease trade between humans. It allows humans to make self-governing rules for collaboration built on immutable codes deployed in the blockchain.

A **Decentralized E-Commerce Website** is a place where sellers may join to offer their goods to a group of customers. A marketplace seller's duty is to connect the proper vendors and customers to generate sales via an amazing multi-vendor platform - suppliers have a location to gain awareness and sell their commodities. In contrast, an online store is a single store that sells its own items online. The firm that owns the website and goods manages all marketing and operations. So in our Solution, we are **utilizing cryptocurrency** as a method of **exchange** for goods via **MetaMask.**

**Workflow:**



The above Flow Diagram Represents the workflow of the **transactions** taking place on the platform **using Blockchain** **technology**. Admin is the one who plays the role of the moderator here. Admin has the ability to add the product to the Platform using the Admin Panel.

The data entered by the admin will be pushed to the **Mongo** **DB**, also **the data will be pushed to the Blockchain using IPFS so that we can see the metadata**. Now once the data is pushed to the chain it will be displayed to the end users on the E-commerce Platform named **NEOSHOP.**

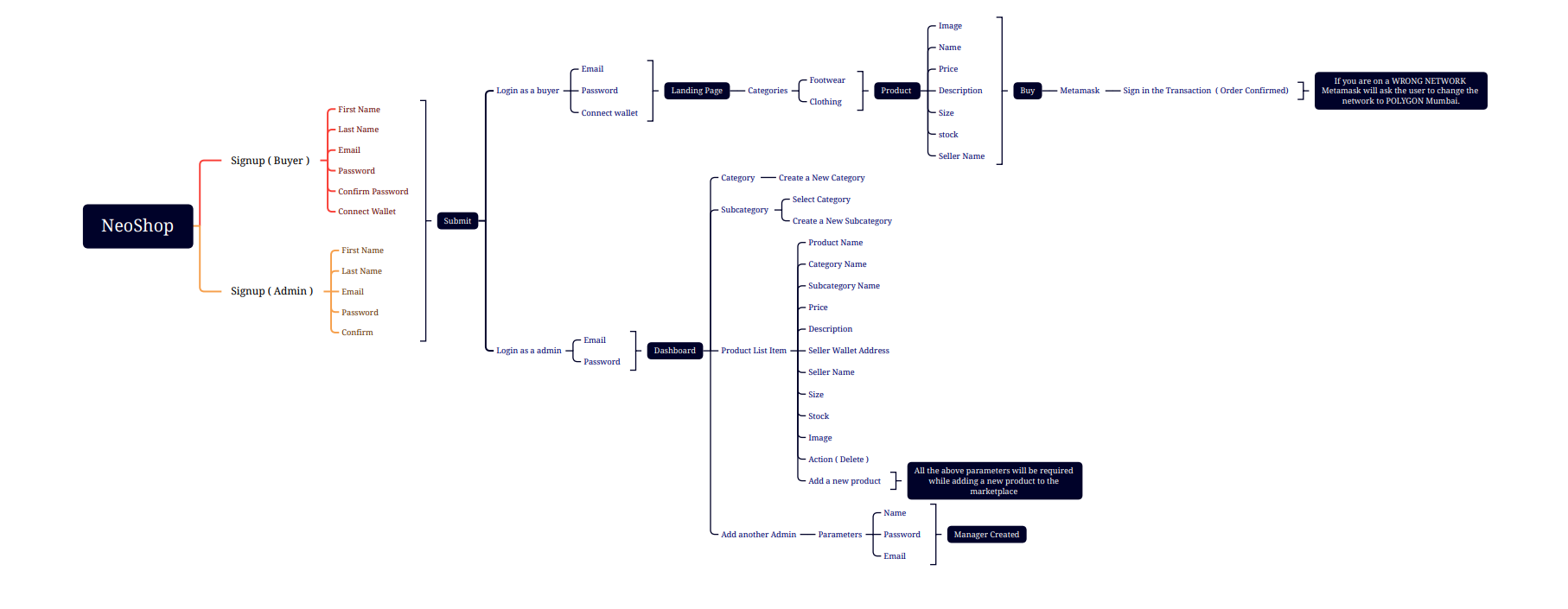
All the orders generated through the website will **be routed through a payment contract** which is deployed on a **POLYGON MUMBAI MATIC chain** & will be used as a **source of Transaction** and thus the payments will be initiated and sent to the retailer’s address.

**Meta Mask is used for signing transactions. All the cryptocurrency transactions are stored on the blockchain.**

**Main Smart Contract Functions Used:**

1. **buy:** This Function is used by the user to buy the product using **MATIC**.
2. **grantRole:** This Function is used by the Admin to give buyer & seller role to specific address, so that they can access the specific functions defined for that role.
3. **payToSeller:** This Function is used by the Admin to send the Amount to the retailer once the product is purchased by the user.
4. **getProductURI:** This Function is used to fetch the metadata of the product which is listed by the Admin.

**Functional Mind Map :**



**UI Figma Design of Ecommerce Portal:**

1. **Landing Page:**

Graphical user interface, application

Description automatically generated

1. **Product Information:**

Graphical user interface, application, website

Description automatically generated

1. **My Orders:**

A screenshot of a computer screen

Description automatically generated with medium confidence

**Wireframing Admin Panel:**

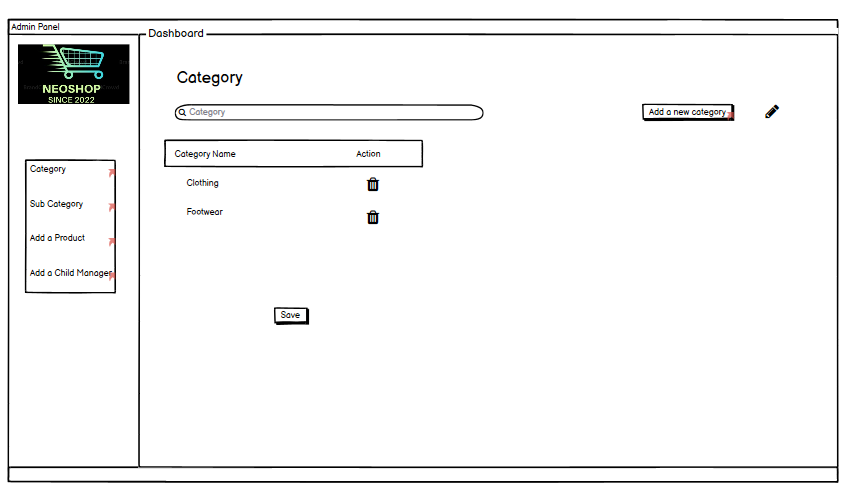
We are giving an ability to the admin to create the complete operations from the panel itself.

**Screen 1: Login**



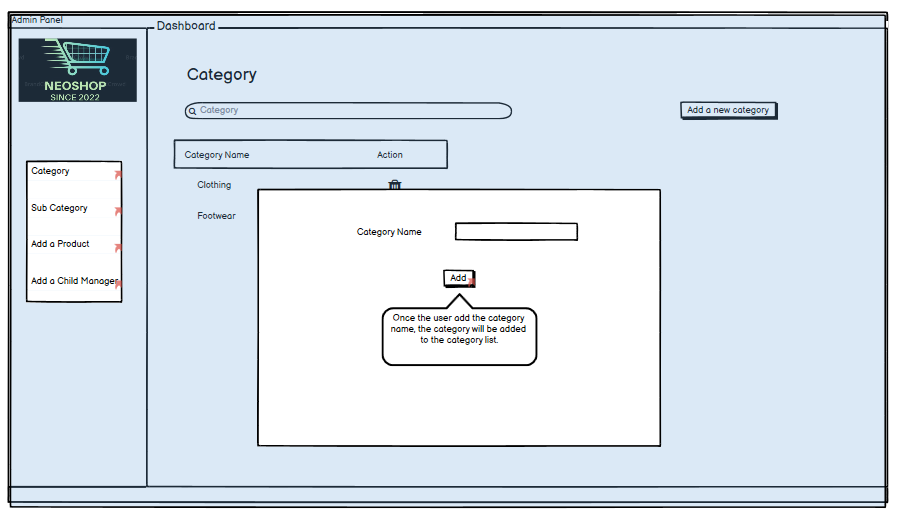
The Admin can log in using the credentials into the panel and can access to all the parameters.

**Screen 2: Category**



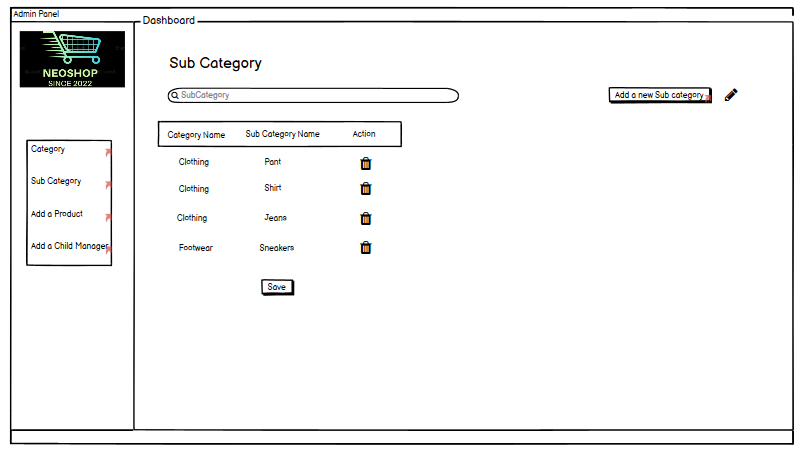
Once the Admin is logged in, he/ she will be redirected to the **category section** wherein the Admin can view the list of categories, also they can create new categories according to their requirement. These Categories will be visible to all the end users while they are browsing the marketplace. Also, the Admin can edit the existing categories listed.

**Screen 3: Add a new Category**



Once the Admin clicks on to **add a new category**, He/ She will be redirected to Add a Category section wherein the admin can create new categories for their user. Once the Admin creates a new category, it will be added to the list.

**Screen 4: Subcategory Category**



Once the Admin creates a category, he/ she will have the ability to **create the Subcategories** under the listed categories.

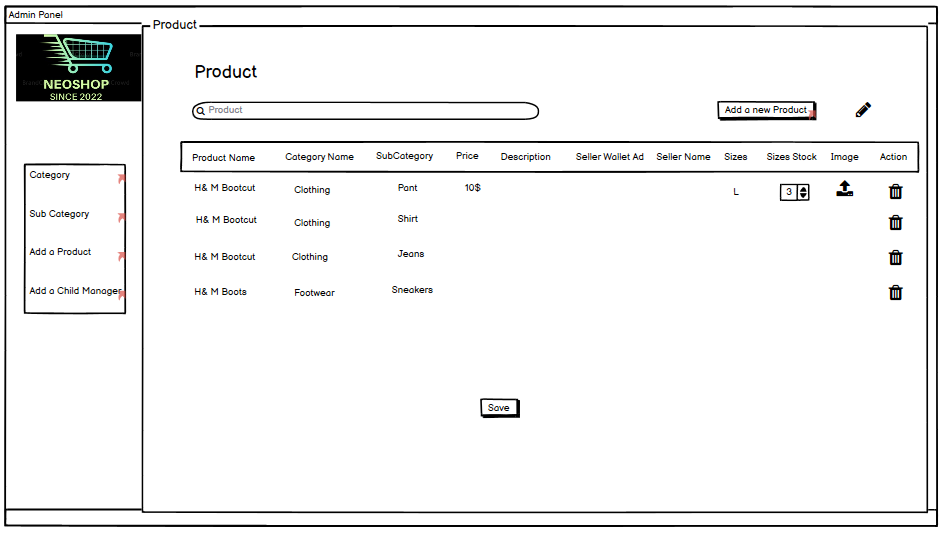
**Screen 5: Add a Subcategory Category**

Graphical user interface

Description automatically generated

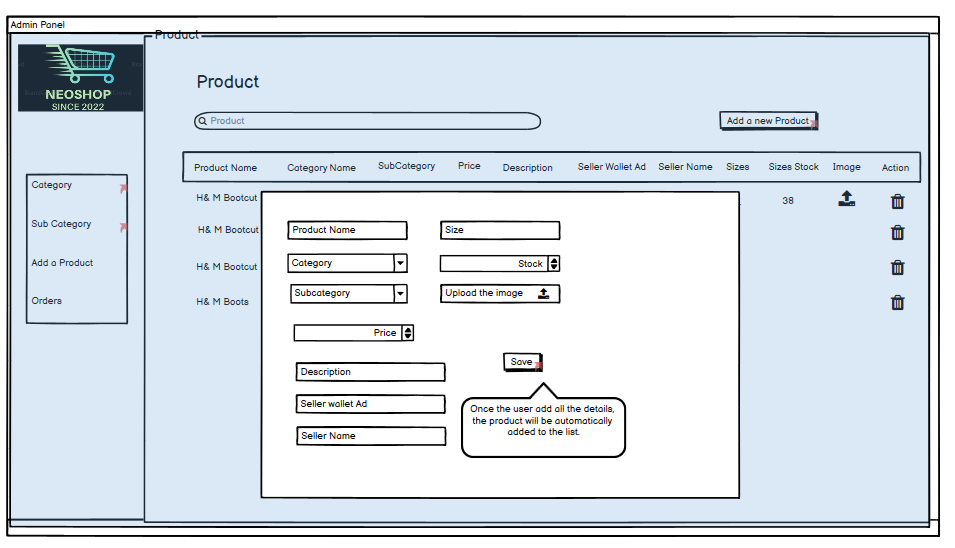
On clicking on the **Add a Category button**, The Admin will be redirected to the page wherein they can choose the category and then **create a subcategory** accordingly. Once the Admin clicks on Add the subcategory will be added to the list.

**Screen 6: Product**



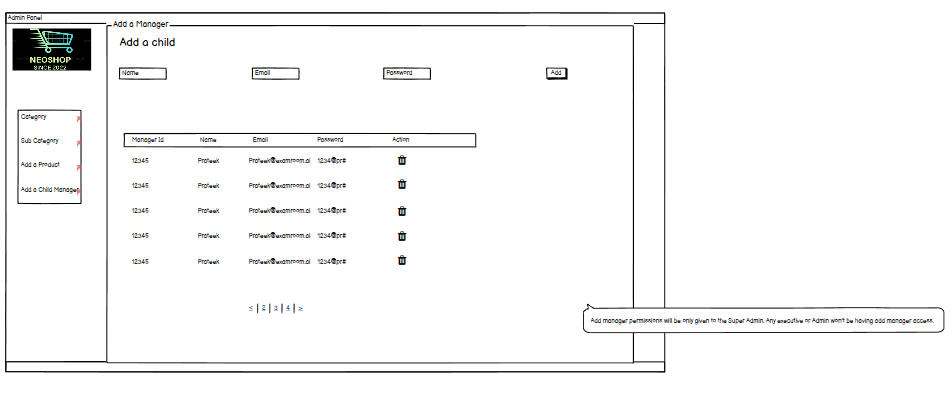
Admin also has the ability to view the **list of products** and also add new products to the list which will be displayed on the marketplace. The product information includes different **parameters** like Product Name, Category Name, Subcategory Name, Price, Description, Seller Wallet Address, Seller Name, Size, Stock, etc. Also, Admin can upload the images of the product from the Admin Panel itself.

**Screen 7: Add A Product**



Admins have the ability to add a **New Product**. Different **parameters** like Product Name, Category Name, Subcategory Name, Price, Description, Seller Wallet Address, Seller Name, Size, and Stock are required while adding a new product to the list.

**Screen 8: Add Another Admin**



**Add another Admin (Manager )** is a functionality wherein the **Super Admin can create credentials for the Admin**. This is mainly used for the operations team to manage the operations. N number of managers can be created who will have access to the panel and **manage all the operations**.

**Tools Used:**

**Mindmap: XMind, DRAW.io**

**Wireframing: Balsamiq, Figma**

**Development: Solidity, React.js, Node.js, Express.Js, Ether.js, MongoDB**