

Assignment 2

DTB : SUPPLY CHAIN MANAGEMENT

[FRONT-END]

Utkarsh Tripathi: 2021202007

Devansh Doshi: 2021202002

Husen Kagdi: 2021201077

As a continuation to the previous assignment, we have implemented a front-end to our already existing smart contract which will handle a supply chain management system with the constraints previously mentioned.

The same has been built using the following tools/technologies:

- Solidity,
- Truffle,
- Ganache,
- React JS,
- Metamask Wallet,
- Web3.js Library.

1. UI WOULD PROVIDE THE FOLLOWING FEATURES:

- a) There's an admin page which can be accessed by the deployer of the smart contract who can then add the major actors in the supply chain management system, which are manufacturers and suppliers.
- b) There's a page for suppliers where they can set the limit for the number of parts they have to offer, and can set the price for the same.
- c) There's a page for manufacturers, where:
 - i. They can check the price for the tyres.
 - ii. Buy a certain amount of tyres.
 - iii. They can place a bid for the car body with the common manufacturer(Vedanta).
 - iv. They can check the bids placed by them and their competitor after the bid has been placed.

- v. They can use the functionality 'make product', which will rebalance the number of parts they have in their shelves to produce corresponding car ids for each car which is their final product and update/increment the number of products they have to offer.
 - vi. The manufacturer can set an offer price for their car.
- d) Customers can register themselves to purchase a car. This has been provided on the home page.
- e) There is a page for customers where:
 - i) They can retrieve the price for a car made available by the manufacturers.
 - ii) They can make payment to purchase a car from a manufacturer of their choice.
 - iii) Retrieve the ID of the car they have purchased.
 - iv) Use the retrieved ID to check whether all the parts used in the car and the car are authentic and belong to the claimed suppliers/manufacturers, and can also check whether the ownership has been transferred to them.
- The actor connects to the website using their account on the metamask wallet. We have used the dummy accounts provided on ganache to use for the actors here. They can then use the functionalities on the website by paying a certain amount of fees by signing the transaction through their wallet. All these functionalities provided are being made available by making use of the smart contract instance which has been generated and making relevant calls to its public methods using web3.js.
- The 3 goals to be achieved are internally managed/achieved by the code written in the smart contract which was deployed and has been connected to the front-end:
 - The bidding will remain secret until both the competing parties have placed their respective bids. It will not be shown on the web-page for either of the participants until the condition has been met.
 - The resource would be distributed optimally to the bidders which has been implemented internally in the code.
 - The customer can go to their section in the website and retrieve their Car ID and using that, they can verify the authenticity and the ownership of the product.