**APS1050 Executive Summary of Personal Project**

1002576722 Ziqi Chen

**What the DApp is:**

The DApp is extended from the pet-shop tutorial with a lot of major modification on frontend UI and backend contract. Compare to the pet-shop tutorial this DApp allows users to pick items prior to checkout and allows users to modify their cart if there is a change of mind. This DApp also allows user to view their prior orders.

**What it does:**

The landing page sends a greeting to the user, and shows the address of the user’s Metamask address.

User can browse products by category, user can add product to cart by simply click on the add button on bottom right corner of each product, users can also modify amount of item in their cart or delete the item form the cart.

There is also an account menu that allows users to navigate to their pervious orders. Purchases details made by this account will be displayed, including total price and a list of purchased items.

Worth mentioning all the items are stored in local storage which means if the page is closed and reopened the cart is preserved. Cart items are stored account-based so if the user switches account the cart will be updated to current account’s cart, but if the user switches back the cart will be resumed. This is same for the orders.

**Why it is relevant:**

The original pet-shop DApp is very basic and doesn’t provide much functionalities, I have made a lot of major feature changes on frontend including the cart and the order history. These features allows this DApp to be used in reality. I also updated the contract to provide public functions createOrder, getOrder and getItem, these functions allows user to publish order and retrieve orders.

In order for smoother experience I used window.ethereum.on(“accountsChanged”, callback) to handle account change event.