TechZilla, LLC

eShop

Phase II: Design Report
For
Online Electronic Shopping

Version 1.2

Prepared By: Team D

Amadou W Jallow

Luigi G Otoya

Sagun Pandey

Pierce K Ruddock Taylor

Hodgson C Tetteh

grade: 98
comments: yet another excellent report top of the class,
except some minor glitchs in col. diag.

Revision History

Date	Version	Description	Author(s)
03/31/2021	1.0	Initial Plan	Team D
04/11/2021	1.1	Added E/R and collaboration diagram	Team D
04/27/2021	1.2	Worked on the pseudocode & UI	Team D

eShop	Version: 1.2
Phase 2: Design Report	Date: 04/27/2021
DI-V-04272021	

Table of Contents
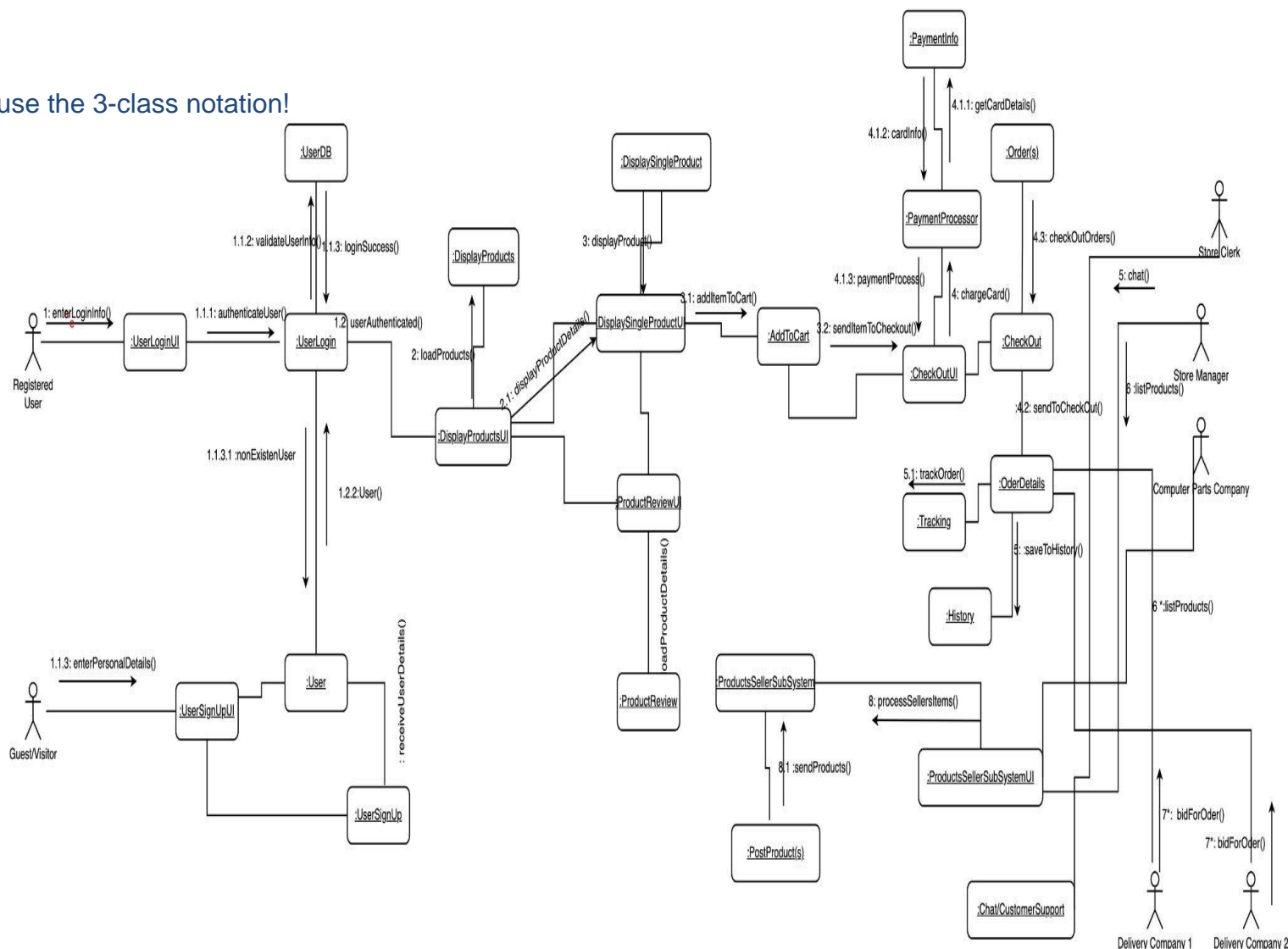
Design Report

1. Introduction

1.1 Purpose

This section includes a description of the webstore TechZilla using various formal and informal diagrams to describe the features and functionality of the system and its parts.

1.2 Collaboration/Communication Class Diagram

use the 3-class notation!

2. Use-Case Scenarios and Diagrams

I.   Use case: Authentication

- **Normal Scenario:**
  - Registered User is asked to enter an email and password that was previously used to register for the webstore.
  - On successful authentication, the page is reloaded, and the registered user is signed into the registered user's account.
- **Exceptional Scenario:**
  - The registered user enters an email or password that is not registered in the system. A message displays the error, and the registered user is prompted to try again.
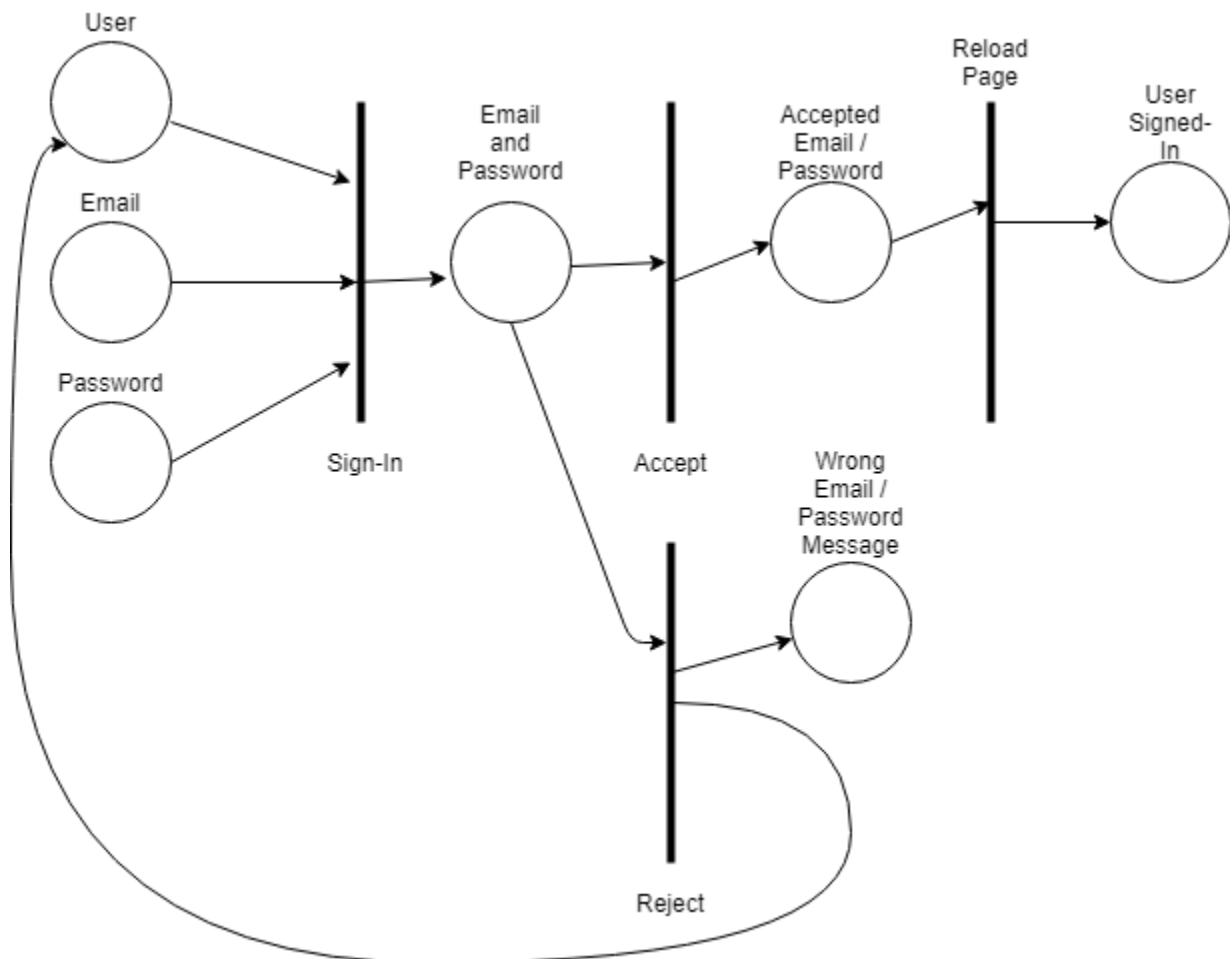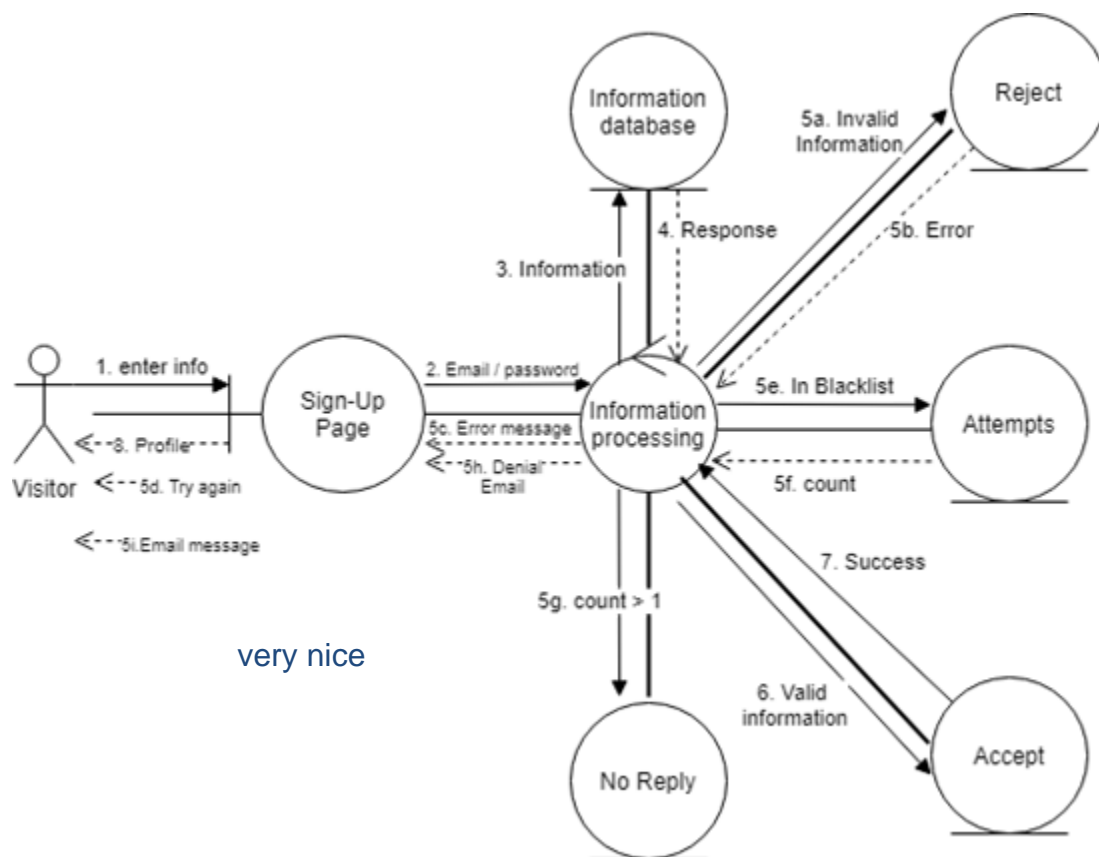
II.   Use Case: Sign-up
- **Normal Scenario:**
  - Visitor wants to become a registered user to the webstore. The visitor is asked to enter an email and password that will register them to the webstore.
  - On success the newly registered user is taken to their profile page in which they can enter personal information such as name, age, address, and credit.
- **Exceptional Scenario:**
  - The visitor enters an invalid email or an email that is already registered in the system. A message displays the error, and the visitor is prompted to re-enter an email and password.
  - If an email address is on the blacklist then a denial message is sent to the email given.

III.   Use Case: Profile

- **Normal Scenario:**
  - Registered user clicks on the profile tab to view details of the registered user's account.
  - The system returns the registered user's profile.
- **Exceptional Scenario:**
  - The registered user loses connection to the webstore in which the system returns access denied.
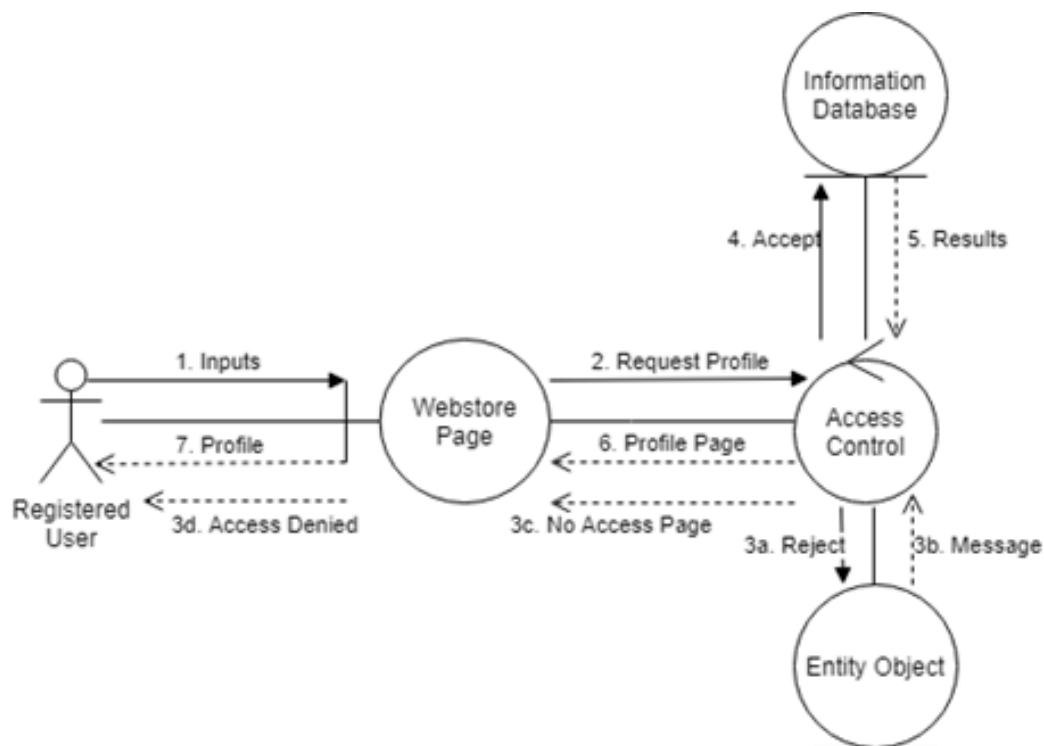
IV.   Use Case: History

- **Normal Scenario:**
  - Registered User requests to view the order history page from profile page.
  - The system returns to the registered user the page which contains the user's complete order history.
- **Exceptional Scenario:**
  - If there is no search History available, the system displays a message saying so.

put numbers on all msgs!

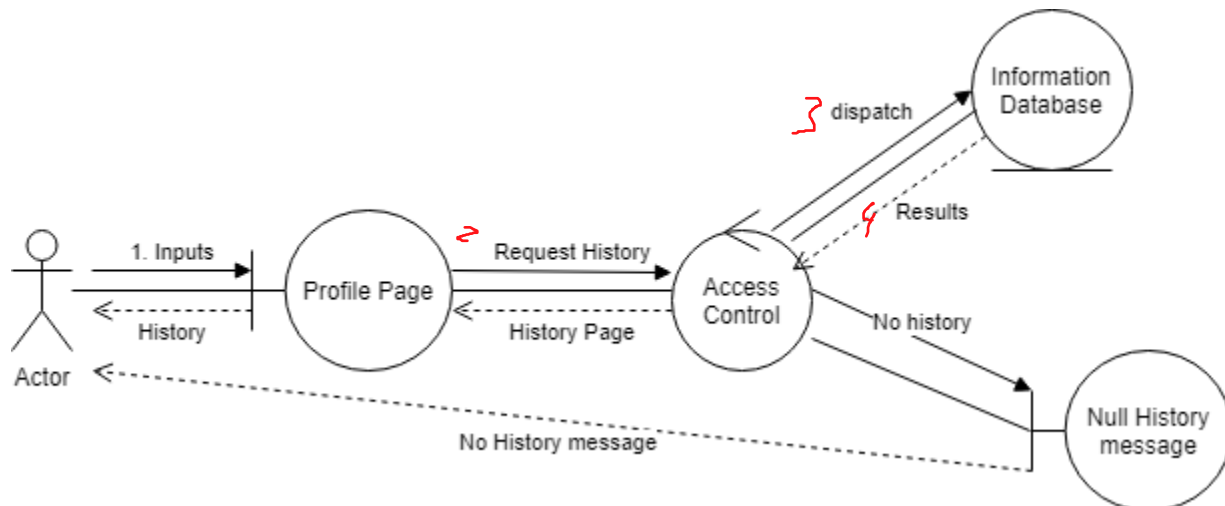## V.   Use Case: Discussion Forum

- **Normal Scenario:**
  - Registered User posts to the discussion Forum over a particular item.
  - Registered User posts to the discussion Forum with a complaint of another customer that is violating rules, such as using inappropriate language.
  - Registered Users can view posts made by other Registered Users.
  - Registered Users can respond to other Registered Users posts.
- **Exceptional Scenario:**
  - A Store Manager can place warnings pertaining to the complaint made by a Registered User. One violation generates one warning.
  - 3 warnings toward a Store Clerk, computer or delivery company suspends them from the webstore.
  - Store Clerk, computer or delivery company can fight against the complaint.
  - The Store Manager decides to issue a warning or not, and informs the parties involved of the justifications.
  - A redacted warning generates an asterisk. Any customer who has their complaint reversed three times is placed on a blacklist. An email will be sent to the Registered User to give a last chance.
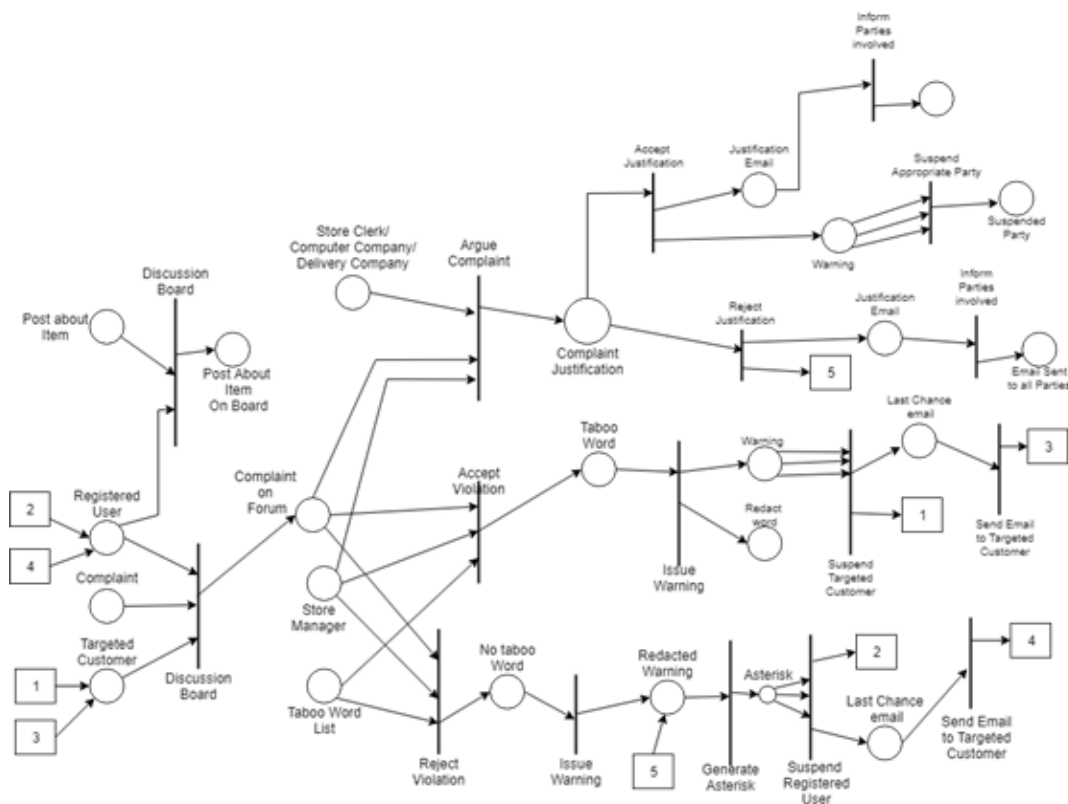
VI.   Use Case: Customer Support

- **Normal Scenario:**
  - Registered users and visitors can interact with customer support. To contact customer support, the registered user/ visitor submits an email to a Store Clerk of the webstore with the respective inquiry.
  - The email is processed by the system and read by a Store Clerk. The Store Clerk formulates a response and submits the response which the registered user/ visitor receives.
- **Exceptional Scenario:**
  - The Store Clerk does not have a response and forwards the email to higher-ups (Store Manager).
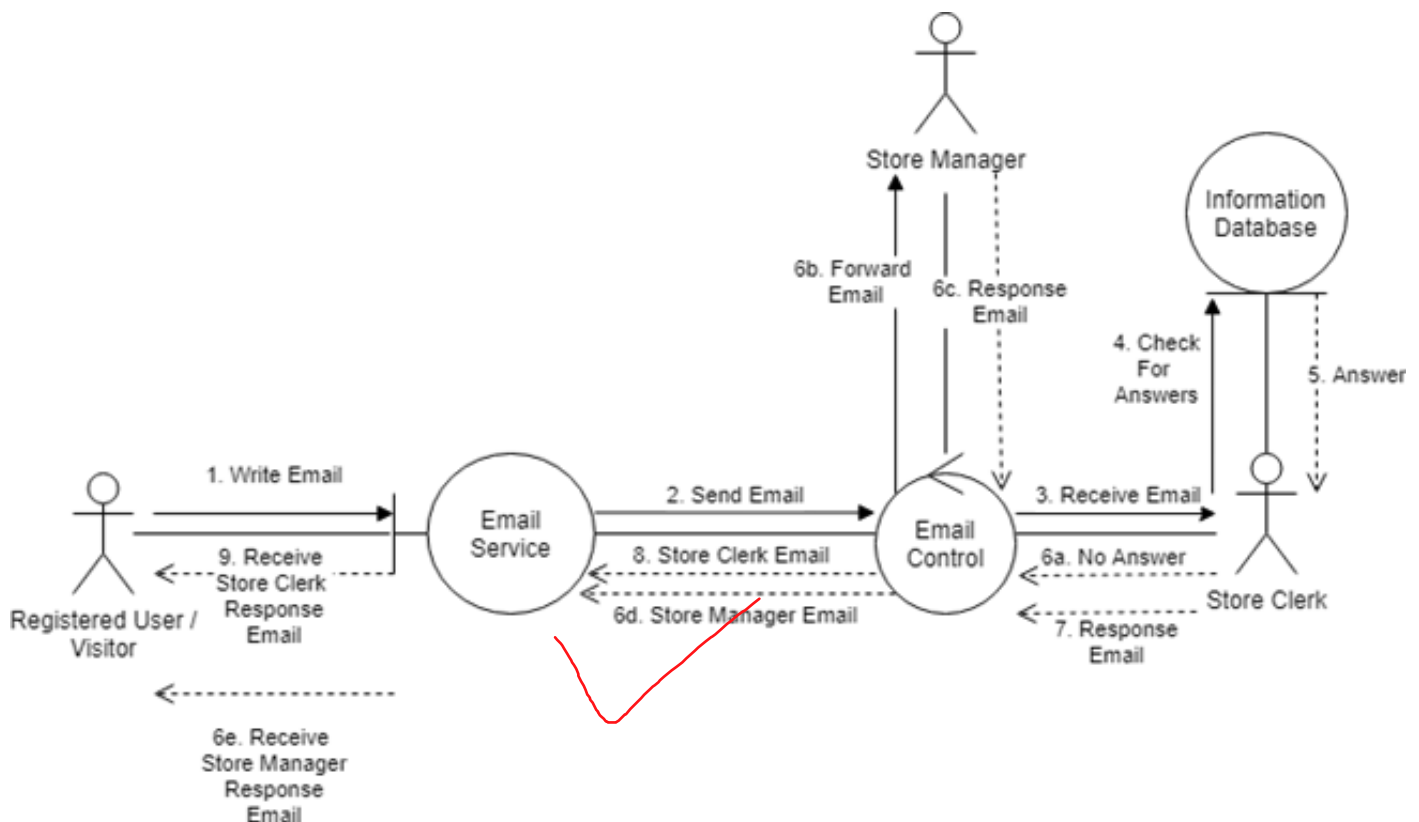
VII.   Use Case: View Products

- **Normal Scenario:**
  - Any user may view the listings available on the webstore.
  - Users may use the search bar or dropdown navigation bar to view the listings of the webstore.
  - The web-store displays the product listings to the user.
- **Exceptional Scenario:**
  - Searched item or item from navigation bar does not exist or is no longer in the system. The system displays a message indicating the error.
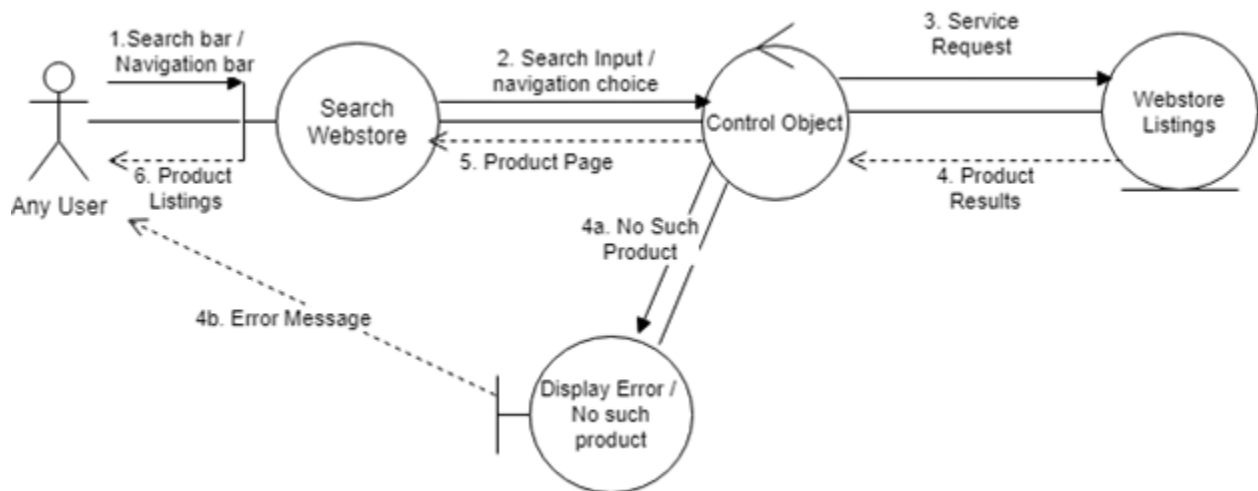
VII.   Use Case: Add to Cart

- **Normal Scenario:**
    - Any user may view the listings available on the webstore.
    - A User may use the search bar or dropdown navigation bar to view the listings of the webstore.
    - The web-store displays the product listings to the user.
- **Exceptional Scenario:**
    - Searched item or item from navigation bar does not exist or is no longer in the system. The system displays a message indicating the error.
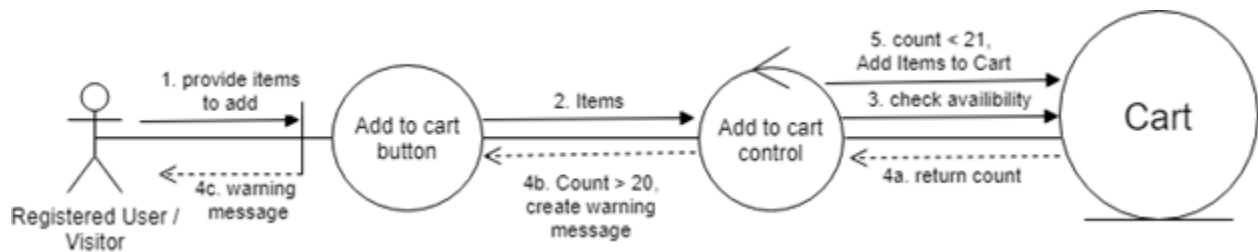
IX.   Use Case: Checkout

- **Normal Scenario:**
  - The Registered user has items in the cart ready for purchase.
  - The Registered user uses the checkout button when ready to complete purchase. Afterwards the system asks the registered user for a payment method to use and for the desired shipping address.
  - On success the system transfers the purchase to the delivery subsystem and the system displays a success message.
- **Exceptional Scenario:**
  - If poor credit or not enough money is available in the account to make a purchase, then the submission is returned with a warning message and the registered user is prompted to enter another payment method.

X.   Use Case: Tracking

- **Normal Scenario:**
    - Registered User has purchased an item from the webstore. The delivery company that has the winning bid provides the registered user the tracking information by placing it the account of the registered user.
- **Exceptional Scenario:**
    - Purchased items are lost in transit. Delivery company issues an email to the registered user and to a store clerk who then issues the registered user a refund.
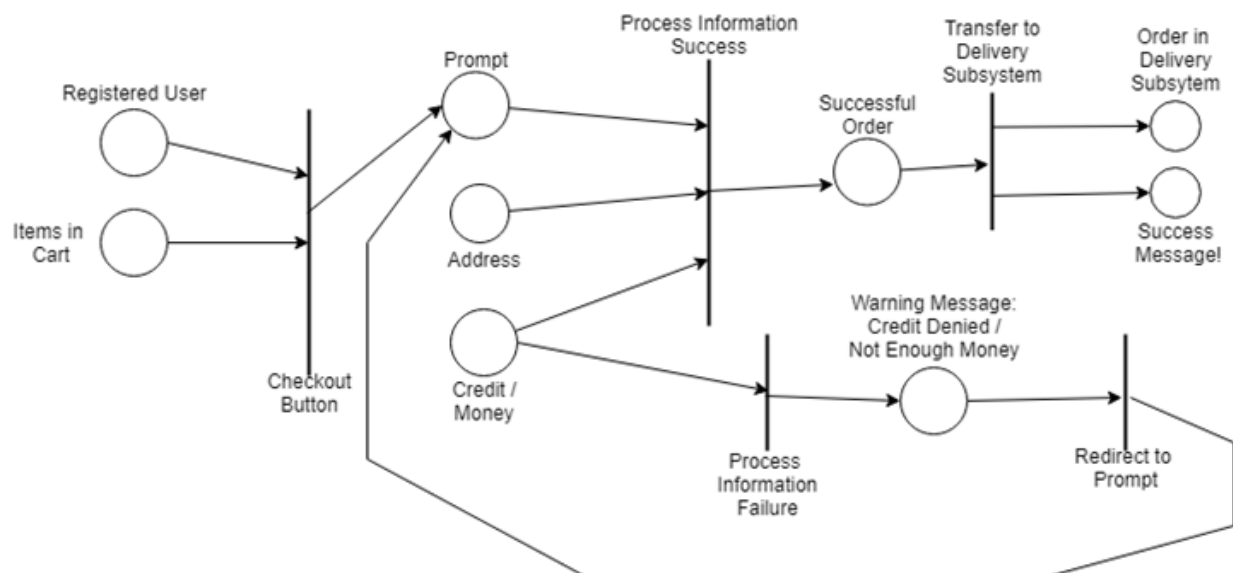
XI.   Use Case: Purchase Item Information

- **Normal Scenario:**
    - Delivery Companies can view the purchase information of an item that was purchased by a registered user. The Delivery company that has the winning bid receives the authorization to package and ship the order.
- **Exceptional Scenario:**
    - The Registered User cancels the order, then the Delivery companies do not bid, and the order cancellation is processed.
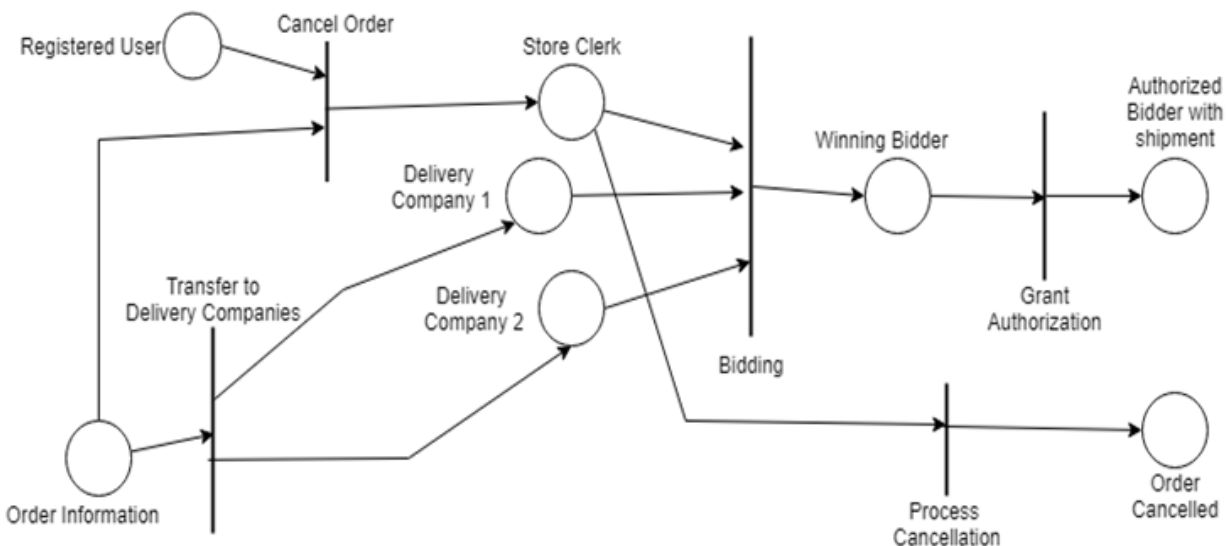
XII.   Use Case: Product(s) Seller system

- **Normal Scenario:**
    - ○ Computer Parts company offers to the Store Manager(s) products they wish to be sold on the webstore.
    - ○ The Store Manager accepts the offer and places the said product(s) onto the webstore.
- **Exceptional Scenario:**
    - ○ The Store Manager declines the offer, and the Computer Parts company makes a new offer in which the Store Manager can reject or accept again.
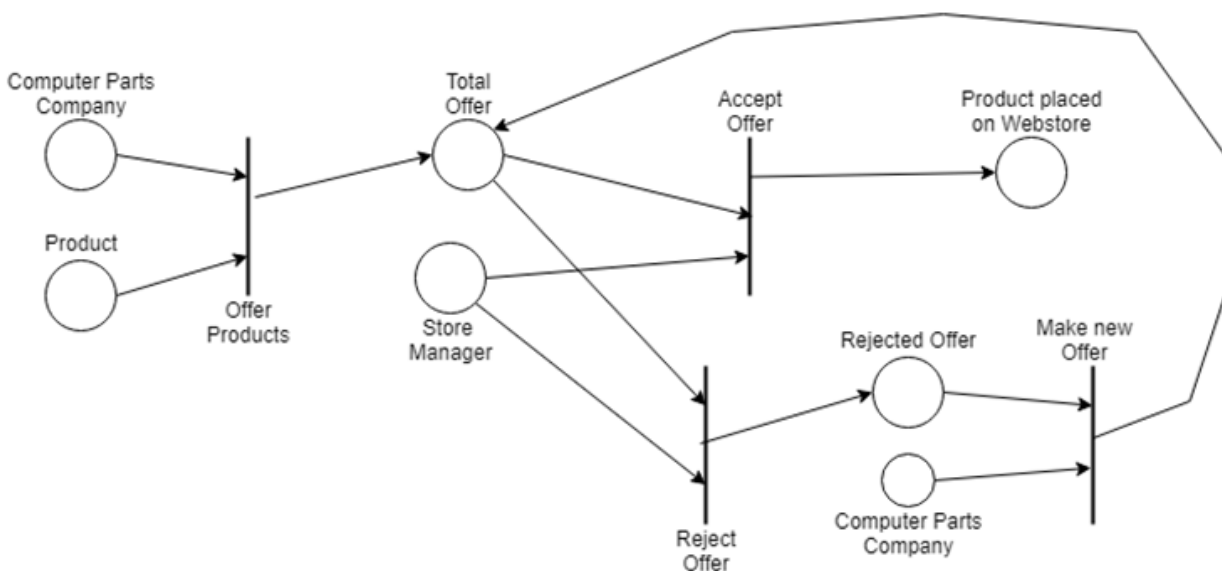
3. E/R Diagram for the Entire System:

fine E/R (side note: the diagramming quality is rather poor, drawing with hand will look better)

I. **Customers**
Can log in buy products and get the products delivered
They can also talk to an employee

II. **Products**
Products are multiple items available in the store
They are supplied buy suppliers and added by the store manager

III. **Employee**
An employee is any staff of the company
They can talk to the customers

IV. **Store Manager**
Store manager is a senior employee who has all the properties of employees but
has the ability to add products to the website

V. **Visitor**
A visitor is anyone who comes to the website to check it out
A visitor can become a customer by signing up

4. Detailed Design:

     As proof of concept, we have created a user-interface prototype for the home page
and login page. We used figma to design these prototypes. Figma is a vector graphics
editor and prototyping tool which is primarily web-based. The following contents are just
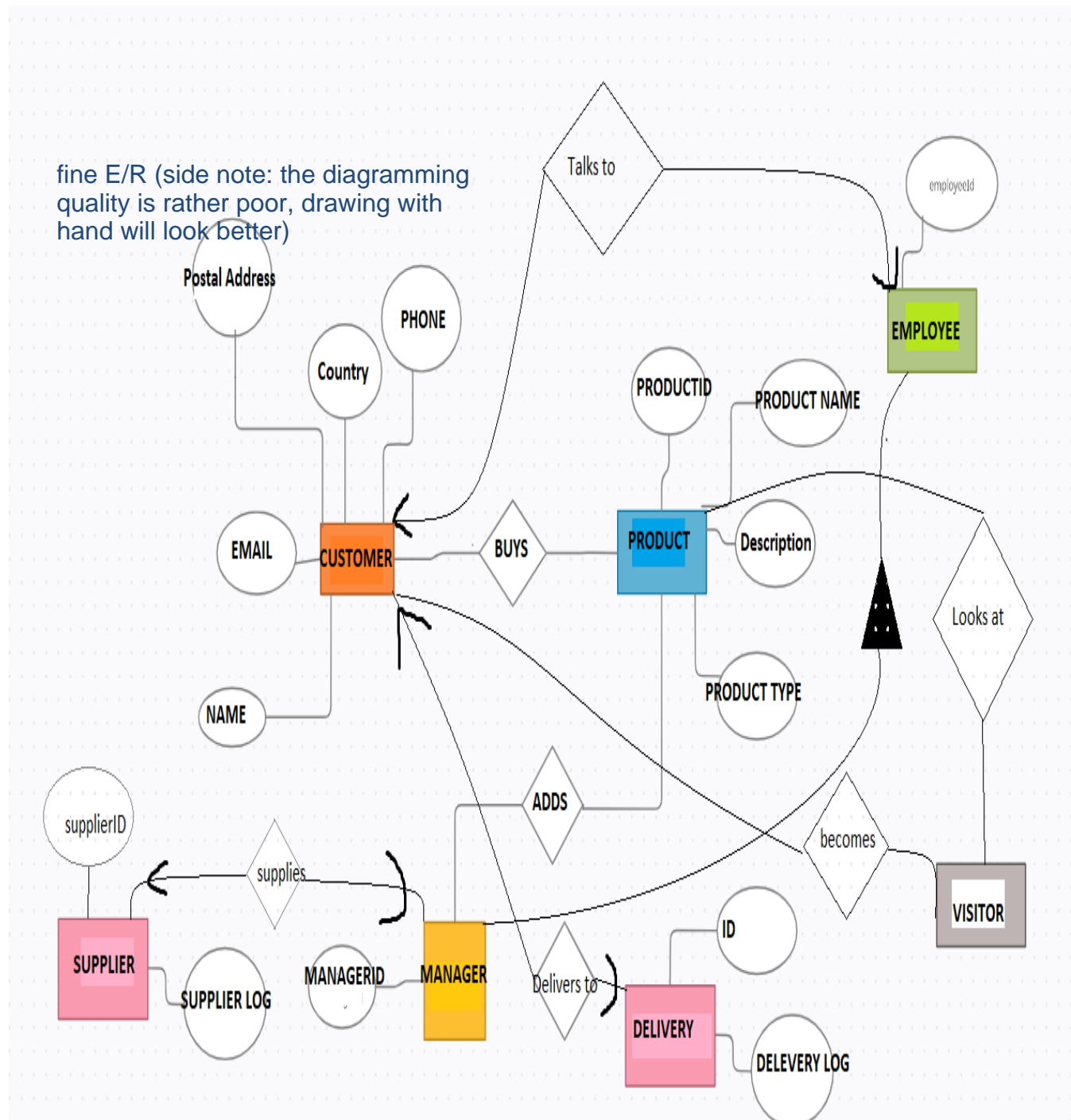a rough draft to show how the store is going to function. It may look different later in the
future. In addition, pseudo code has been provided below for all methods within each
class.

I. **Homepage /HeroSection**
This section is the main page of our online web application which is a gateway to
use our system fully.
function HeroSection()

```
import React from 'react';
import '../App.css';
import './HeroSection.css';


function HeroSection() {
  return (
```

```jsx
    <div className='hero-container'>
        <video src='video-2.mp4' autoPlay loop muted />
      <h1>Welcome to techZilla</h1>
      <p>Your trusted online Computer World</p>
    </div>
  );
}


export default HeroSection;
```

function Home()

```jsx
import React from 'react';
import '../../App.css';
import Cards from '../Cards';
import HeroSection from '../HeroSection';
import Footer from '../footer/Footer';



function Home() {
  return (
    <>
      <HeroSection />
      <Cards />
      <Footer />
    </>
  );
}


export default Home;
```

II.   **Navigation bar with Login and Sign Up**
In the navigation bar, we have login and signup toggles which will give you access to become a loyal member of the store. We are thinking of adding something called guest, through which customers can shop anonymously.

```
function Navbar() {

    const [click, setclick] = useState(false);

    const handleClick = () =>{
        setclick(!click)
    }
    const closeMobileMenu = () => setclick(false);
```

III.   Sign Up

<u>Receive:</u> User's information such as name, email, username and password
<u>Return:</u> Success/Already exists - Login/ failed

```
export default class SignUp extends Component {
    render() {
        return (

            <div className="signup">

                <div className="auth-wrapper">
                <div className="auth-inner" style={{position:
'absolute', left: '50%', top: '50%',transform: 'translate(-50%,
-50%)'}}>
                    <form>
                        <div className="login-sign-up-nav">
                            <Link className="nav-link current"
to='/signup'>Sign up</Link>
                            <Link className="nav-link"
to='/login'>Login</Link>
                        </div>

                        <div className="form-group">
                            <TextField label="First name"
type="text" fullWidth/>
                        </div>
```

```jsx
                          <div className="form-group">
                              <TextField label="Last name" type="text"
fullWidth/>
                          </div>

                          <div className="form-group">
                              <TextField label="Email" type="email"
fullWidth/>
                          </div>

                          <div className="form-group">
                              <TextField label="Username" type="text"
fullWidth/>
                          </div>

                          <div className="form-group">
                              <TextField label="Password"
type="password" fullWidth/>
                          </div>

                          <br></br>

                          <Link className="btn btn-danger btn-block"
to='/signup'>Sign up</Link>
                          <p className="forgot-password text-right">
                              Already registered? <Link
to='/login'>Login</Link>
                          </p>
                      </form>
```

IV.   Login

<u>Receive:</u> User's credentials such as username and password

<u>Return</u>: Logged in or failed

```jsx
export default class Login extends Component {
```

```
    render() {
        return (
            <div className="login" >

                <div className="auth-wrapper" >
                <div className="auth-inner" style={{position:
'absolute', left: '50%', top: '50%',transform: 'translate(-50%,
-50%)'}}>
                    <form>
                        <div className="login-sign-up-nav">
                            <Link className="nav-link"
to='/signup'>Sign up</Link>
                            <Link className="nav-link current"
to='/login'>Login</Link>
                        </div>

                        <div className="form-group">
                            <TextField label="Username" type="text"
fullWidth/>
                        </div>

                        <div className="form-group">
                            <TextField label="Password"
type="password" fullWidth error id="standard-error-helper-text"
                                    defaultValue="Hello World"
                                    helperText="Incorrect
password."/>
                        </div>

                        <br></br>

                        <Link className="btn btn-danger btn-block"
to='/'>Login</Link>
                        {/*<p className="forgot-password
text-right">
```

```
                            Forgot <a href="#">password?</a></p>*/}
                </form>
```

V.   Cards

This section holds the most popular items to be shown in the front page which
will attract our users to dive into the system.

```
function Cards() {
  return (
    <div className='cards'>
      <h1>View our popular Items</h1>
      <div className='cards__container'>
        <div className='cards__wrapper'>
          <ul className='cards__items'>
            <CardItem
              src='logo1.png'
              text='Desktops/PC'
              label='Shop now'
              path='/services'
            />
```

VI.   CardItems/new pages

This section is the main part of our store. It holds the inventory.

```
function CardItem(props) {
  return (
    <>
      <li className='cards__item'>
        <Link className='cards__item__link' to={props.path}>
          <figure className='cards__item__pic-wrap'
data-category={props.label}>
            <img
              className='cards__item__img'
              alt='Computers'
              src={props.src}
            />
          </figure>
          <div className='cards__item__info'>
```

```
            <h5 className='cards__item__text'>{props.text}</h5>
        </div>
      </Link>
    </li>
  </>
  );
}


export default CardItem;
```

VII. Footer

Stores company's information, social media links and copyright license tag and to navigate to the top of the home page.

```
function Footer(){
    return (
        <div className='footer-container'>
            <section className='footer-start'>
{/*            <p className='footer-title'>

            </section>
            <div className='footer-links'>
                <div className='footer-link-wrapper'>
                    <div className='footer-link-items'>
                        <h2>COMPANY INFORMATION</h2>
                        <Link to='/'>About techZilla</Link>
                        <Link to='/'>Contact</Link>
                        <Link to='/'>Hours of Operation</Link>
                    </div>
                    <div className="footer-link-items">
                        <h2>CUSTOMER SERVICE</h2>
                        <Link to='/'>Frequently Asked
Questions</Link>

                        <Link to='/'>Check Order Status</Link>
                        <Link to='/'>Terms of Service</Link>
```

```
                    </div>
                    <div className='footer-link-items'>
                        <h2>SOCIAL MEDIA</h2>
                        <Link to='/'><i className='fab fa-twitter'/>
Twitter</Link>
                        <Link to='/'><i className='fab fa-youtube'/>
Youtube</Link>
                        <Link to='/'><i className='fab
fa-facebook'/> Facebook</Link>
                        <Link to='/'><i className='fab
fa-instagram'/> Instagram</Link>
                    </div>
                </div>
            </div>
            <section className='website-copyright'>
                <small>&#169;2021 techZilla LLC. All rights
reserved</small>
            </section>
        </div>
```

5. System Screens:

The following are the initial GUI screens of the system users can access. Here, users can create an account using any email address. A confirmation email will be sent to their email address. If the email address is used already, you can log in to your previous account by clicking Login here.

Homepage:

This is the main page of our Computer Store. On Click of "Login" or "Sign Up", users will be directed to join. After that, users will be asked to input some information based on what they click.
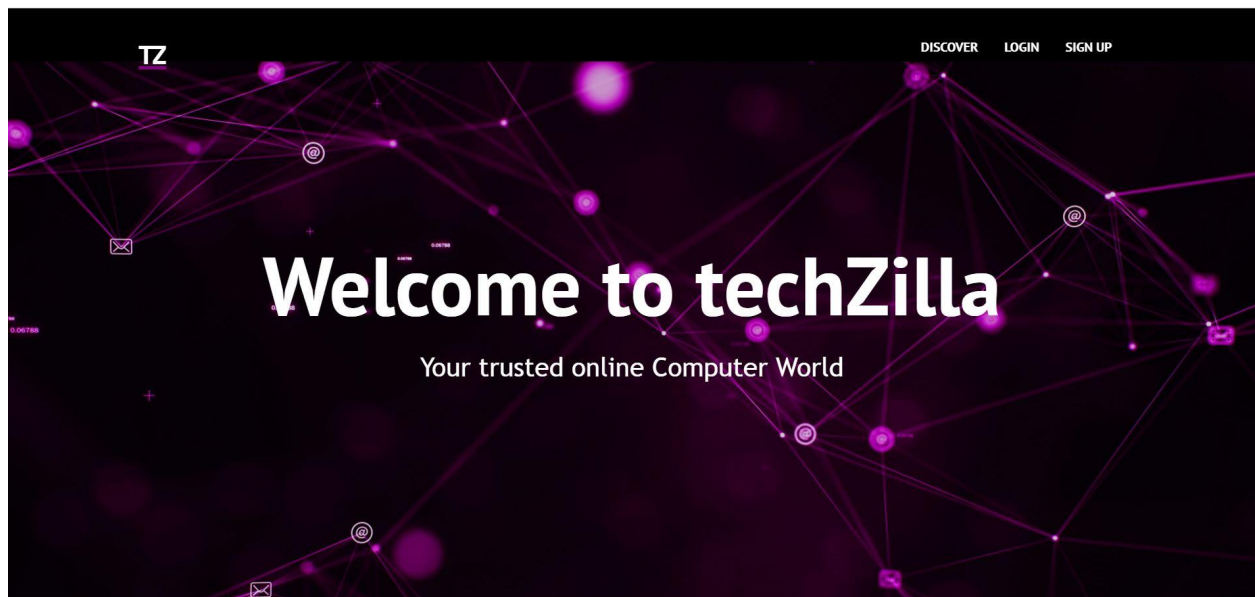
Figure: Main page of the techZilla computer store Team D

Signup:

  Here, users can create an account using their full name, any email address. A confirmation email will be sent to their email address. If the email address is used already, you can log in to your previous account by clicking Login here.

Figure: Sign Up box of the techZilla computer store Team D

Login:

  When the "Log in here" is clicked, users will be directed to the Login page as shown below. Users can enter their email addresses and password to login into their eshop account. If the password doesn't match, you have an option to click on "forgot password" which will send you a password reset link to your email address. By clicking the link from your email, users can easily reset their passwords.

Figure: Login box of the techZilla computer store Team D

Togglebox with popular items:
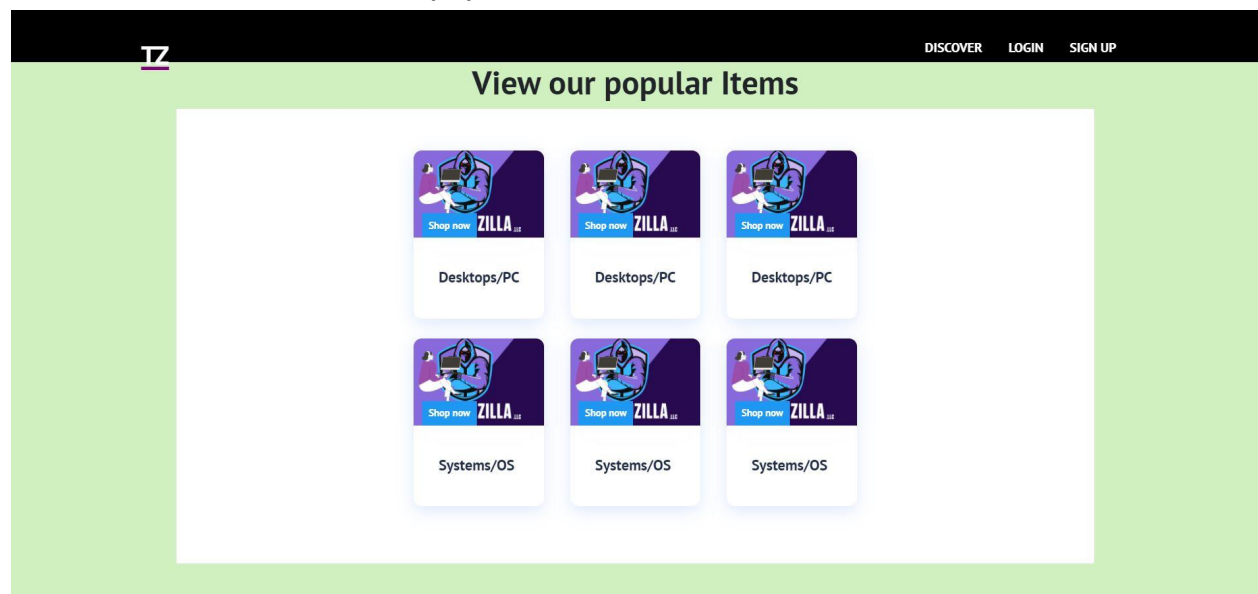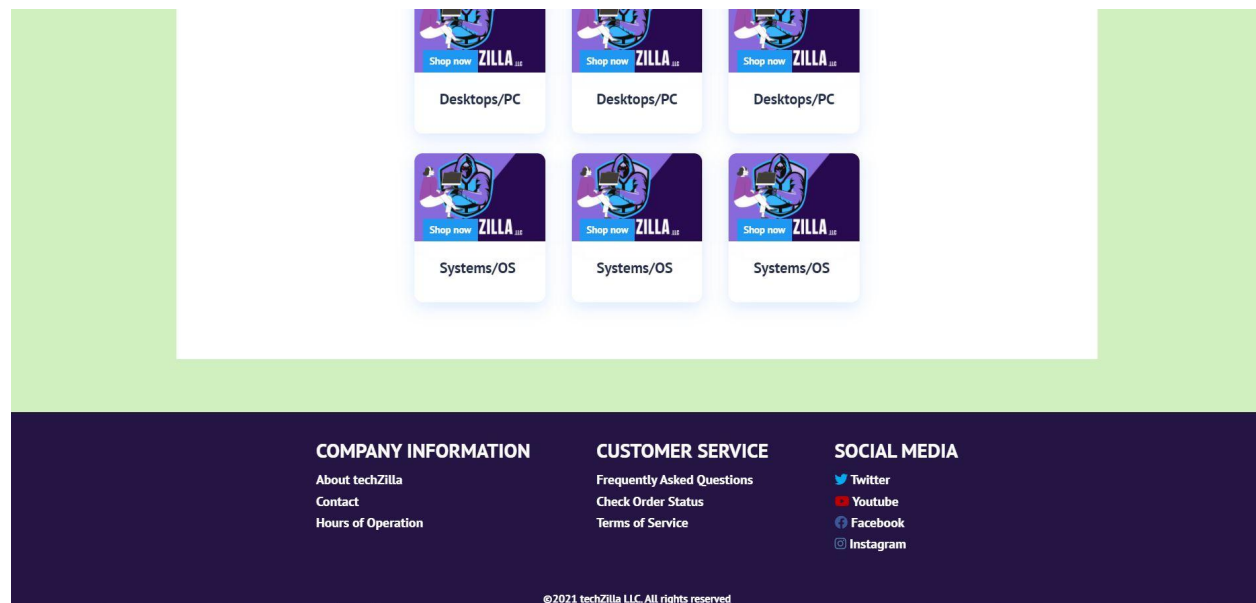
This section holds the 6 main popular items .

Figure: 6 most popular items of the techZilla computer store Team D

Footer:

eShop	Version: 1.2
Phase 2: Design Report	Date: 04/27/2021
DI-V-04272021	

       This section stores company's information, customer service, all our company's social media links and copyright license tag and to navigate to the top of the home page.

Assets page:

This page contains lists of all the computers and accessories which are sold by the company. We have added features to filter our search. Some of the features are: pricing range, operating systems, ratings, purpose and architecture.
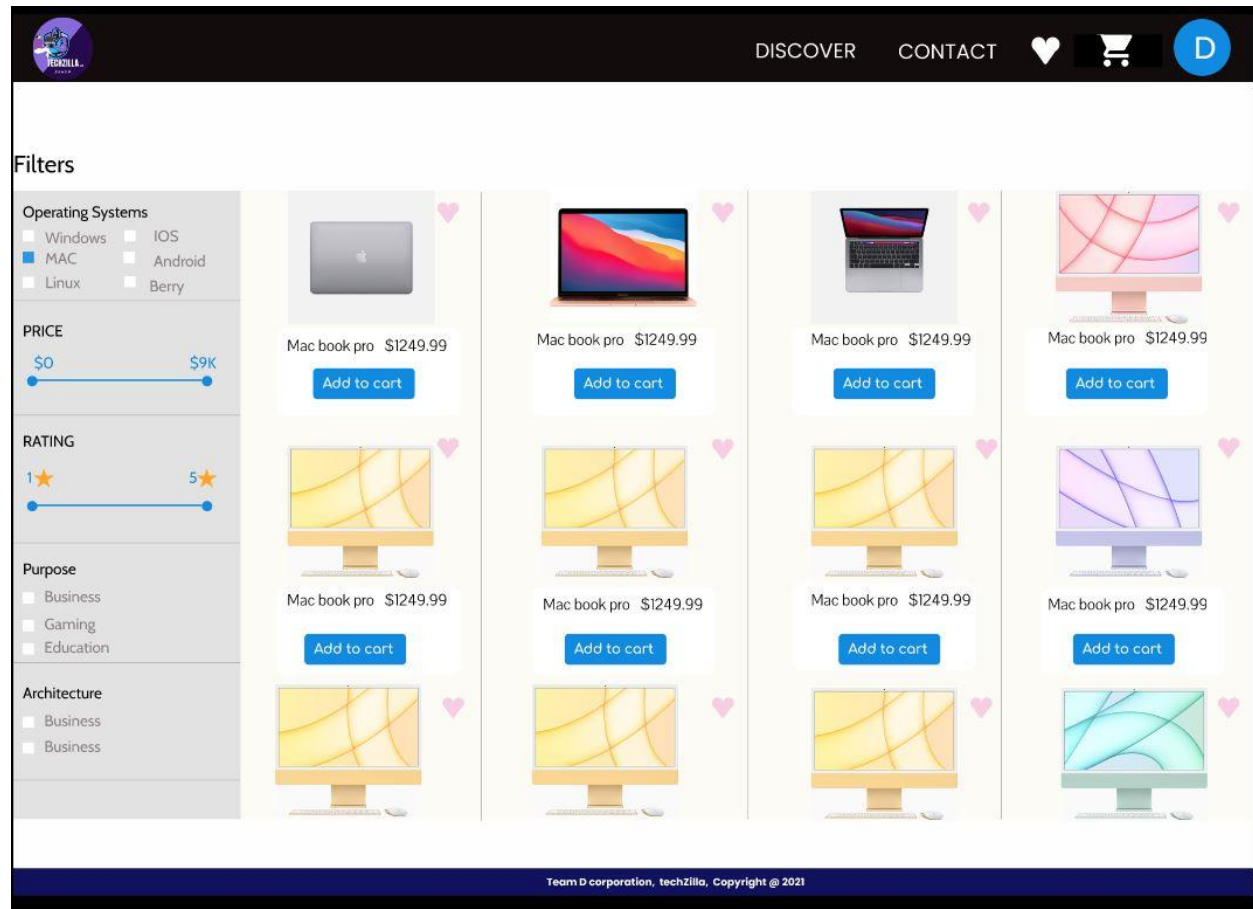
Figure: computers and accessories of the techZilla computer store Team D

6. Minutes of Group Meetings

We have allocated weekly google meet meetings and discord platforms to discuss our progress and next steps.

Date	Length	Meeting Discussion	Goals for the Next Meeting
02/25	1 hr.	Introduced to each other/ discussed some ideas for the project	Discuss frameworks/ language, development process
03/30	1 hr.	Discussed on the specs posted by the professor and brainstormed ideas	Discuss on using react and installation process
03/31	1 hr.	Downloaded dependencies,created react app, and published on git-repo	Brainstorm on the overall design and work on the phase I report
04/01	1 hr.	Discussed Phase I report	Continue working on the report
04/02	2 hrs.	Divided work to each individual and started working on the project	Prototyping on figma and continue working on the project
04/18	1 hr.	Continued working on our own parts	Discuss on the dev. process
04/21	1 hr.	Discussed on the git commands	Discuss on the phase II report
04/25	2 hrs.	Discussed Phase II report	Continue working on the report
04/27	2 hrs.	Worked on the E/R, petri-nets diagram of the entire system and overall report	Start working on the new features of the project

7. Git Repository:

Here is the address of the git repository of our team's work so far:
https://github.com/sagunpandey123/TechZilla