National University of Sciences & Technology

School of Electrical Engineering and Computer Science

Department of Computing

CS 213 Advance Programming (3+1): BSCS-7AB Fall 2019

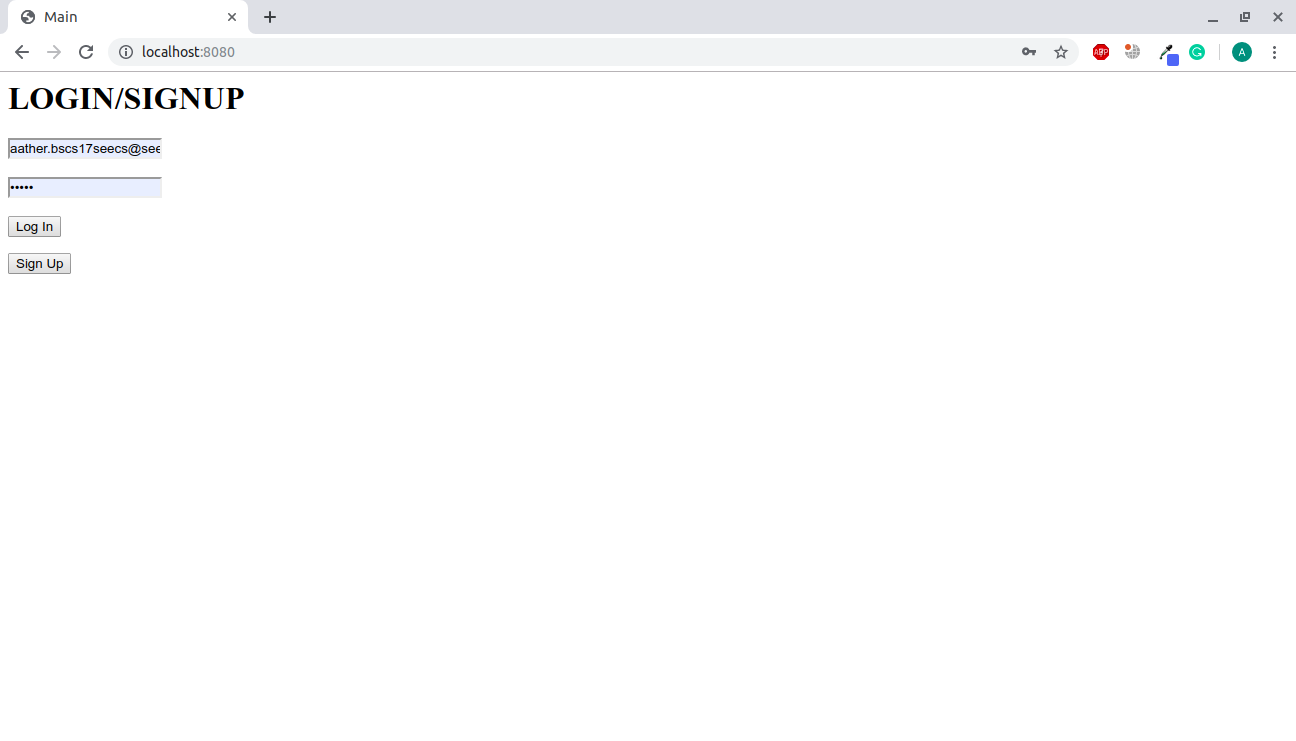
|  |  |
| --- | --- |
| Assignment 1: CRUD and AMS in Express.Js | |
| **CLO 3: Develop applications and tools using various frameworks** | |
| Maximum Marks: 10 | Instructor: Dr. Sidra Sultana |
| Announcement Date: 15th Oct., 2019 | Due Date : 23rd Oct., 2019 5:00 pm on LMS |
| **Name: Ali Ather/ 211226** | **Section: BSCS-7A** |

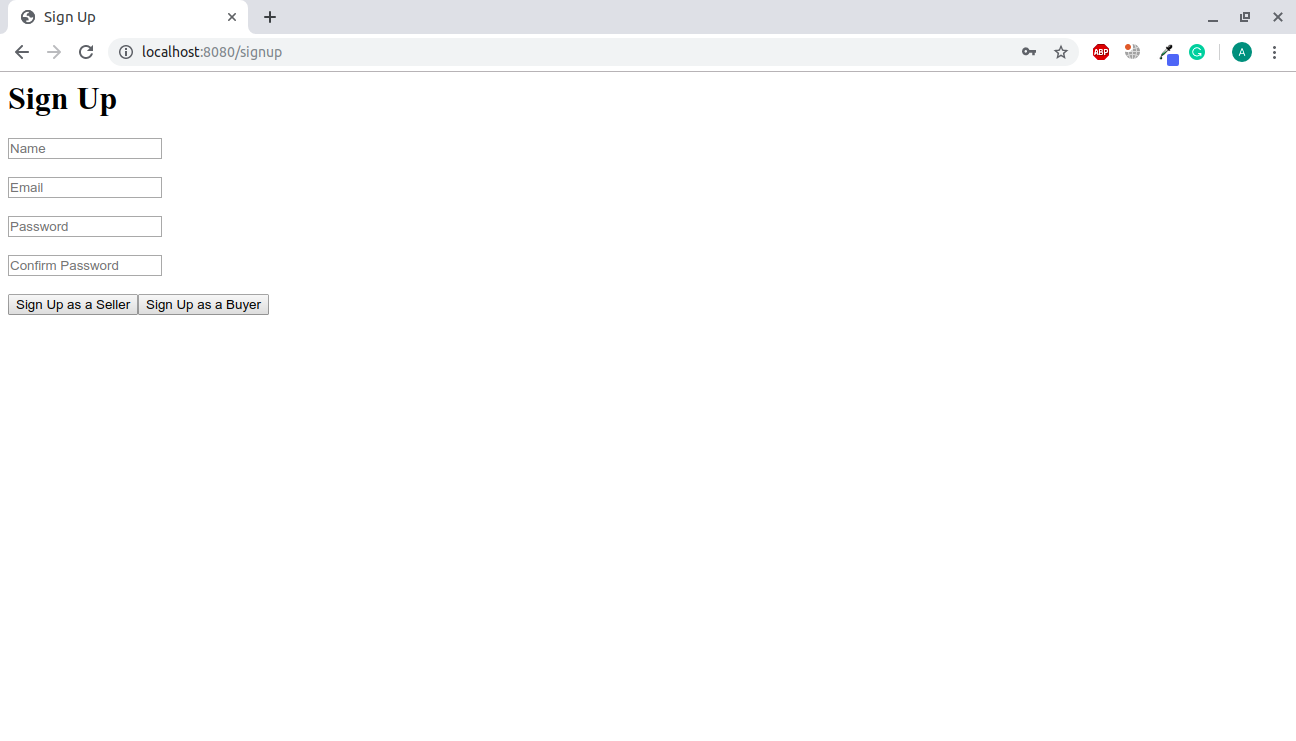
**Scenario**

Code Product manager with name, desc, price, category, image and reviews where only authentic user (the one who owns the product) can add, update and delete product. Admin can add users and assign roles as saler and buyer.

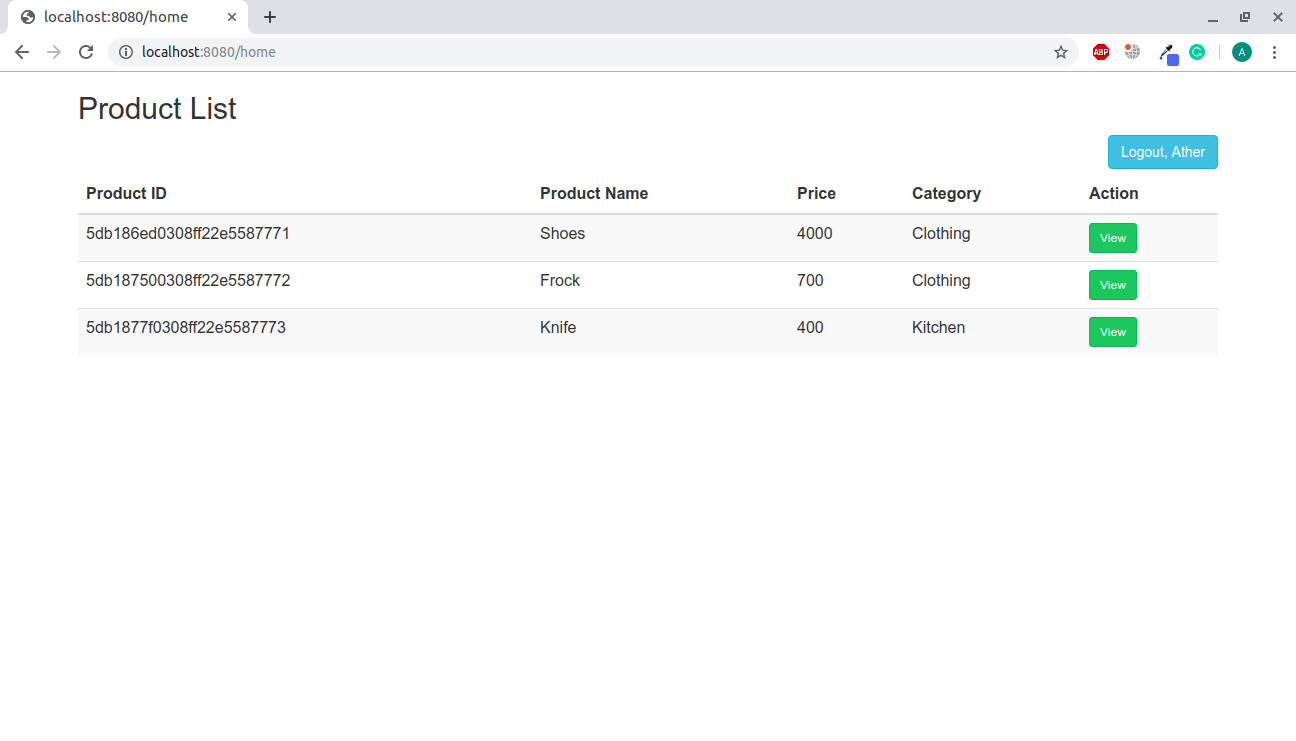
Sample screenshots with lesser fields are given as an example.

**Login and Signup:**

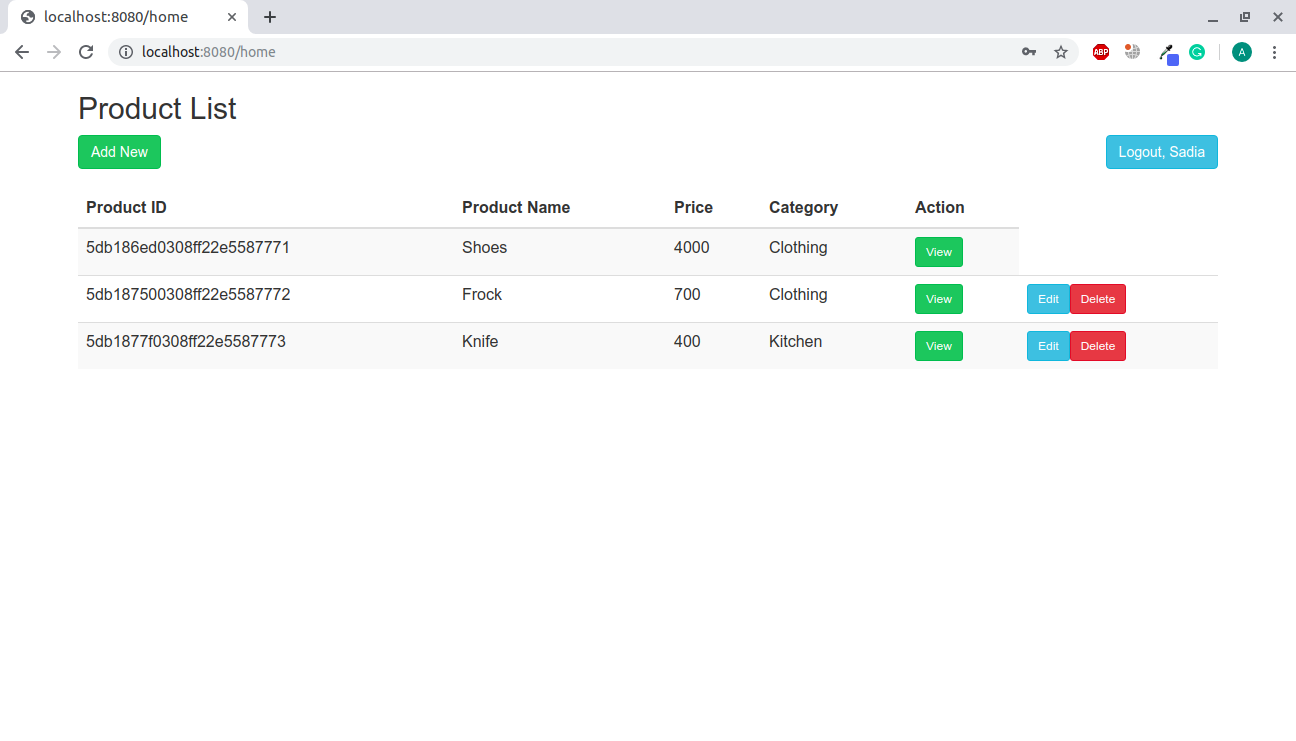




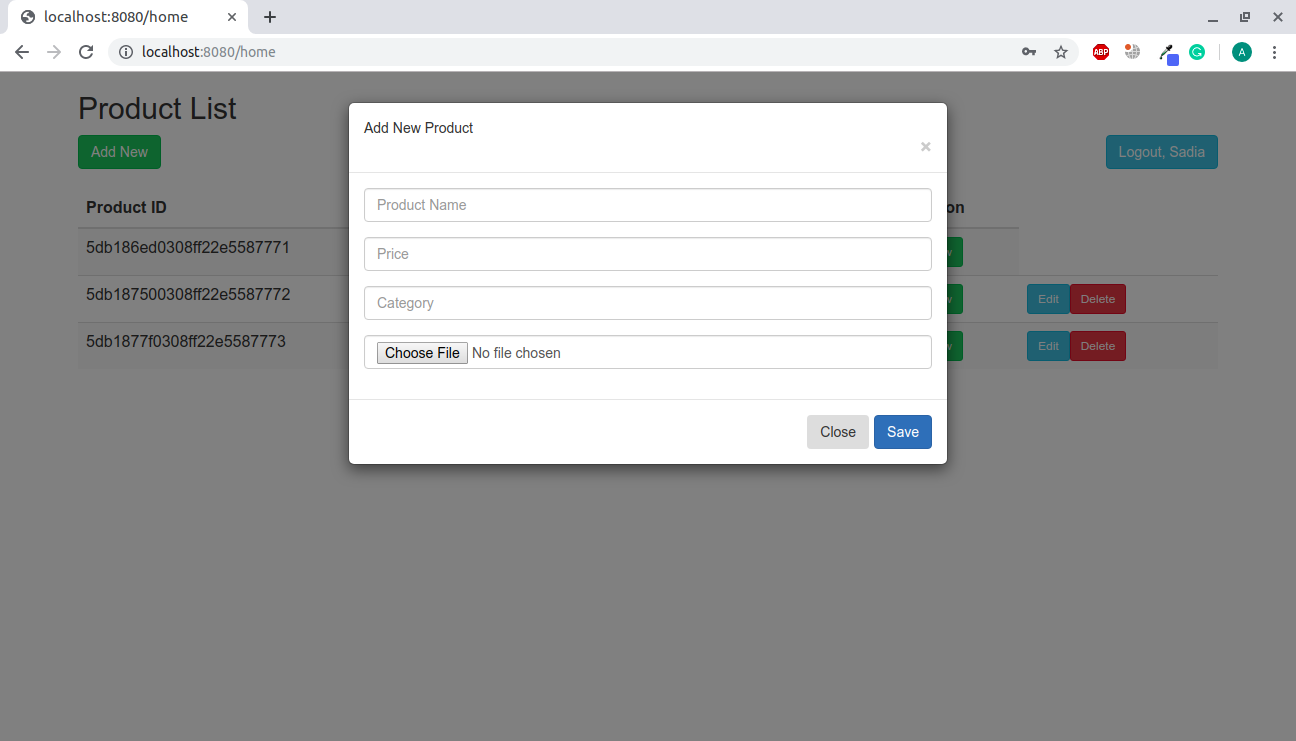
**Buyer:**

****

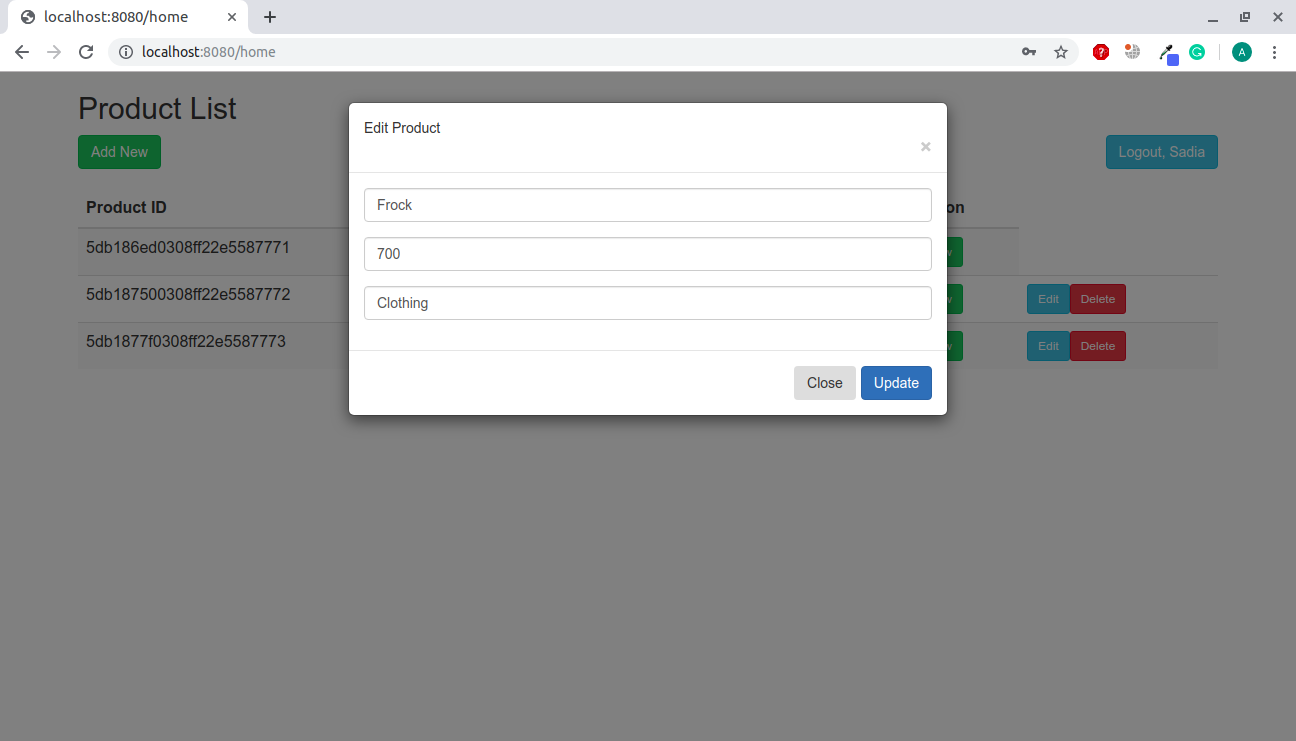
**Seller:**

****

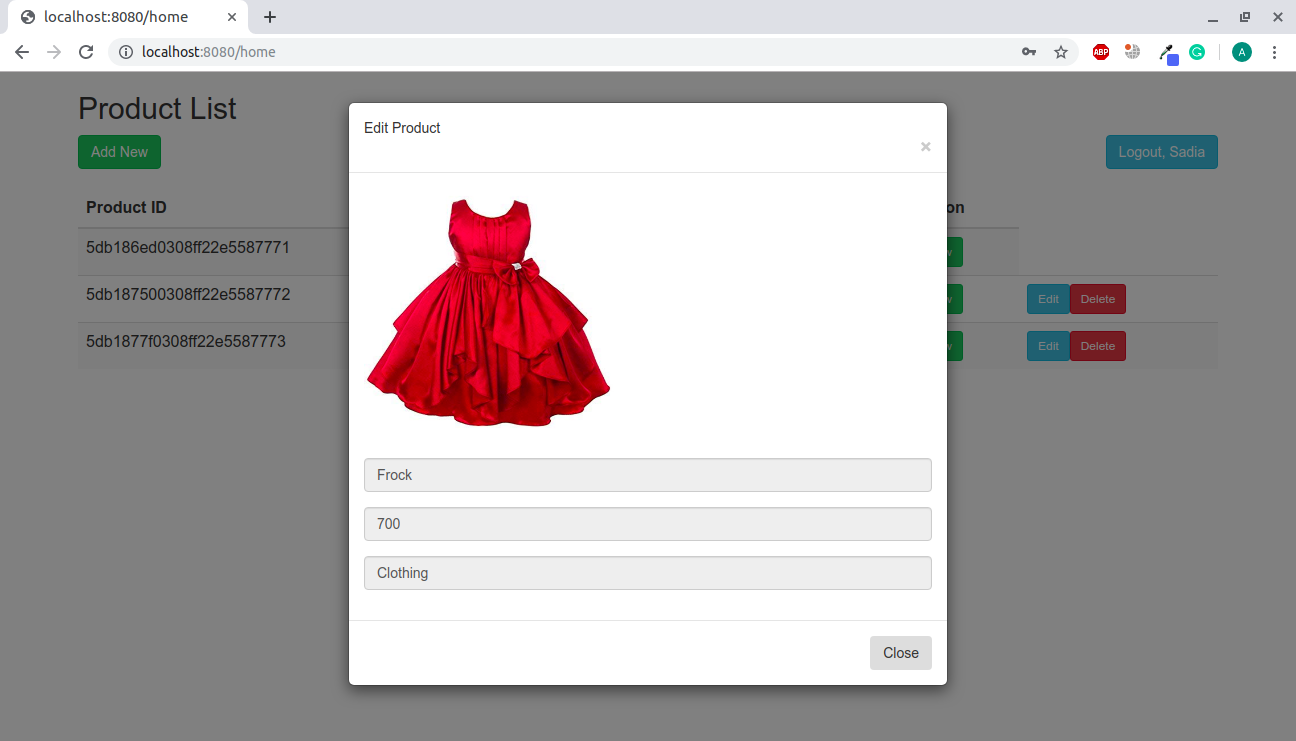
**Add New:**

****

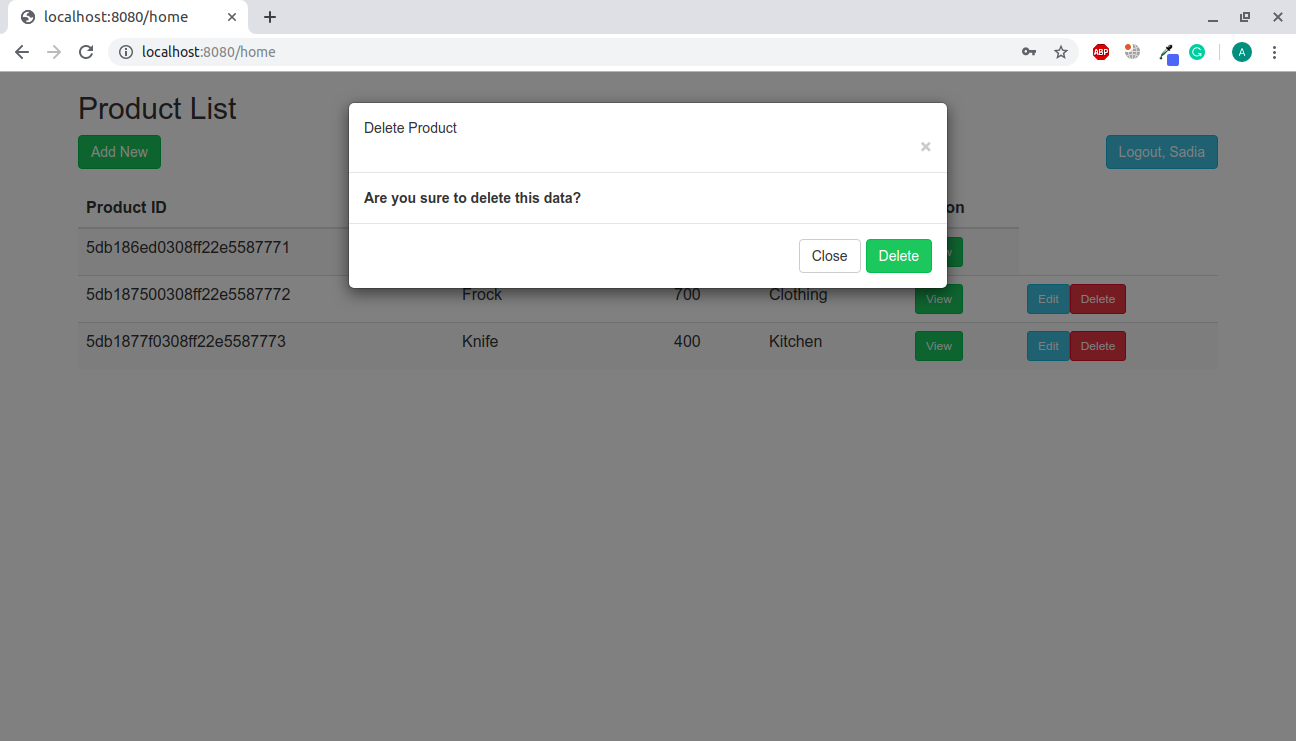
**Edit:**

****

**View:**

****

**Delete:**

****

**Code:**

**Server.js:**

var express = require('express')

var app = express()

var bodyParser = require('body-parser')

var session = require('express-session');

var user\_session;

const multer = require('multer');

const storage = multer.diskStorage({

    destination: function (req, file, cb) {

      cb(null, \_\_dirname + '/public/images')

    },

    filename: function (req, file, cb) {

      cb(null, file.originalname)

    }

  })

const upload = multer({storage: storage});

app.use(express.static( \_\_dirname + '/public'));

var mongoose = require('mongoose');

mongoose.connect('mongodb://localhost/product-manager');

var userSchema = mongoose.Schema({ name: String, email: String, password: String, role: String });

var Users = mongoose.model("Users", userSchema)

var productschema = mongoose.Schema({ name: String, price: Number, category: String, img: String, seller\_id: String });

var Product = mongoose.model("Product", productschema);

app.use(bodyParser.json())

app.use(bodyParser.urlencoded({ extended: true }))

app.use(session({secret: 'secret'}));

app.set('view engine', 'pug')

app.set('views', './views')

app.get('/', function (req, res) {

    res.render('Log in')

})

app.post('/', function (req, res) {

    user\_session = null;

    var userinfo = req.body

    console.log(userinfo)

    if (userinfo.password != userinfo.cpassword)

        res.render('Sign up', { message: "passwords do not match", type: "error" })

    else {

        if (userinfo.seller == 'Sign Up as a Seller')

            var userrole = 'seller'

        else

            var userrole = 'buyer'

        var newUser = new Users({ name: userinfo.name, email: userinfo.email, password: userinfo.password, role: userrole });

        newUser.save(function (err, Users) {

            if (err) {

                res.render('Sign up', { message: "Database error", type: "error" });

            } else {

                res.render('Log in', { message: "Sign up Successful!", type: "success" });

                console.log(req.body)

            }

        })

    }

})

app.all('/signup', function (req, res) {

    res.render('Sign up')

})

app.post('/home', function (req, res) {

    var userinfo = req.body

    Users.findOne({ email: userinfo.email, password: userinfo.password }, function (err, response) {

        if (err || response == null) {

            res.render("Log in", { message: "Invalid Id or password!", type: "error" })

        }

        else {

            user\_session= req.session;

            user\_session.user = response;

            Product.find(function(err, response\_1){

               res.render("home", { user: user\_session.user, products: response\_1 });

            })

        }

    })

})

app.get("/home", function(req, res){

    Product.find(function(err, response\_1){

        res.render("home", { user: user\_session.user, products: response\_1 });

     })

})

app.post("/save",  upload.single('photo'), function(req,res){

    var pro\_info = req.body;

        console.log(req.file.filename);

        var newProduct = new Product({ name: pro\_info.name, price: pro\_info.price, category: pro\_info.category, img: "/images/"+ req.file.filename, seller\_id: user\_session.user.\_id });

        newProduct.save(function (err, Person) {

            if (err) {

                res.render("home", { message: "Error adding Product!", type: "error" })

            }

            else {

                res.redirect("home");

                console.log(req.body);

            }

        })

})

app.post("/update", function(req,res){

    var pro\_info = req.body;

        Product.findOneAndUpdate({ \_id : pro\_info.id  },{name: pro\_info.name, price: pro\_info.price, category: pro\_info.category},function(err, response\_1){

            res.redirect("home");

         })

})

app.post("/delete", function(req,res){

    var pro\_info = req.body;

        Product.findByIdAndRemove(pro\_info.id,function(err, response\_1){

            res.redirect("home");

         })

})

app.listen(8080)

**Home.pug:**

head

  meta(name='viewport' content='width=device-width, initial-scale=1')

  link(rel='stylesheet' href='https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/css/bootstrap.min.css')

  script(src='https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js')

  script(src='https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/js/bootstrap.min.js')

.container

  h2 Product List

  if (type == "error")

    h2(style = "color: red") #{message}

  if(user.role == "seller")

    button.btn.btn-success#addbutton(data-toggle='modal' ) Add New

  button.btn.btn-info(style = "float: right;")#logout Logout, #{user.name}

  br

  br

  table#mytable.table.table-striped

    thead

      tr

        th Product ID

        th Product Name

        th Price

        th Category

        th Action

    tbody

      each product, i in products

        tr

            td #{ product.\_id}

            td #{ product.name }

            td #{ product.price }

            td #{product.category}

            td

              a.btn.btn-sm.btn-success.view(style = href='javascript:void(0);' data-id= product.\_id data-product\_name=product.name data-product\_price=product.price data-product\_category=product.category data-product\_image = product.img) View

            if(user.\_id == product.seller\_id)

                td

                    a.btn.btn-sm.btn-info.edit(href='javascript:void(0);' data-id= product.\_id data-product\_name=product.name data-product\_price=product.price data-product\_category=product.category ) Edit

                    a.btn.btn-sm.btn-danger.delete(href='javascript:void(0);' data-id= product.\_id) Delete

// Modal Add Product

form(action='/save' method='post' enctype="multipart/form-data")

  #myModalAdd.modal.fade(tabindex='-1' role='dialog' aria-labelledby='exampleModalLabel' aria-hidden='true')

    .modal-dialog(role='document')

      .modal-content

        .modal-header

          h5#exampleModalLabel.modal-title Add New Product

          button.close(type='button' data-dismiss='modal' aria-label='Close')

            span(aria-hidden='true') &times;

        .modal-body

          .form-group

            input.form-control(type='text' name='name' placeholder='Product Name' required='')

          .form-group

            input.form-control(type='text' name='price' placeholder='Price' required='')

          .form-group

            input.form-control(type='text' name='category' placeholder='Category' required='')

          .form-group

            input.form-control(type="file" accept="image/\*" name="photo" required)

        .modal-footer

          button.btn.btn-secondary(type='button' data-dismiss='modal') Close

          button.btn.btn-primary(type='submit') Save

// Modal Update Product

form(action='/update' method='post')

  #EditModal.modal.fade(tabindex='-1' role='dialog' aria-labelledby='exampleModalLabel' aria-hidden='true')

    .modal-dialog(role='document')

      .modal-content

        .modal-header

          h5#exampleModalLabel.modal-title Edit Product

          button.close(type='button' data-dismiss='modal' aria-label='Close')

            span(aria-hidden='true') &times;

        .modal-body

          .form-group

            input.form-control.product\_name(type='text' name='name' placeholder='Product Name' required='')

          .form-group

            input.form-control.product\_price(type='text' name='price' placeholder='Price' required='')

          .form-group

            input.form-control.product\_category(type='text' name='category' placeholder='Category' required='')

        .modal-footer

          input.product\_id(type='hidden' name='id' )

          button.btn.btn-secondary(type='button' data-dismiss='modal') Close

          button.btn.btn-primary(type='submit') Update

// Modal Delete Product

form#add-row-form(action='/delete' method='post')

  #DeleteModal.modal.fade(tabindex='-1' role='dialog' aria-labelledby='myModalLabel' aria-hidden='true')

    .modal-dialog

      .modal-content

        .modal-header

          h5#myModalLabel.modal-title Delete Product

          button.close(type='button' data-dismiss='modal' aria-label='Close')

            span(aria-hidden='true') &times;

        .modal-body

          strong Are you sure to delete this data?

        .modal-footer

          input.form-control.product\_id2(type='hidden' name='id' required='')

          button.btn.btn-default(type='button' data-dismiss='modal') Close

          button.btn.btn-success(type='submit') Delete

// Modal View Product

form(action='/view' method='post')

  #ViewModal.modal.fade(tabindex='-1' role='dialog' aria-labelledby='exampleModalLabel' aria-hidden='true')

    .modal-dialog(role='document')

      .modal-content

        .modal-header

          h5#exampleModalLabel.modal-title Edit Product

          button.close(type='button' data-dismiss='modal' aria-label='Close')

            span(aria-hidden='true') &times;

        .modal-body

          img(src="", alt="", height = "250px", width = "250px" )

          br

          br

          .form-group

            input.form-control.product\_name(type='text' name='name' placeholder='Product Name' required='' disabled="disabled")

          .form-group

            input.form-control.product\_price(type='text' name='price' placeholder='Price' required='' disabled="disabled")

          .form-group

            input.form-control.product\_category(type='text' name='category' placeholder='Category' required='' disabled="disabled")

        .modal-footer

          input.product\_id(type='hidden' name='id' )

          button.btn.btn-secondary(type='button' data-dismiss='modal') Close

script.

    document.getElementById("logout").onclick = function () {

        location.href = "/";

    };

script(src='https //ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js')

script(src='https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js')

script.

    $("#addbutton").on('click',function(){

            $('#myModalAdd').modal('show');

    })

    //showing data to edit modal

    $('#mytable').on('click','.edit',function(){

    var id = $(this).data('id');

    var name = $(this).data('product\_name');

    var price = $(this).data('product\_price');

    var category = $(this).data('product\_category');

    console.log(name);

    $('#EditModal').modal('show');

    $('.product\_name').val(name);

    $('.product\_price').val(price);

    $('.product\_category').val(category);

    $('.product\_id').val(id);

    })

  //showing delete record modal

    $('#mytable').on('click','.delete',function(){

    var product\_id = $(this).data('id');

    $('#DeleteModal').modal('show');

    $('.product\_id2').val(product\_id);

    });

  //showing delete record modal

    $('#mytable').on('click','.view',function(){

    var id = $(this).data('id');

    var name = $(this).data('product\_name');

    var price = $(this).data('product\_price');

    var category = $(this).data('product\_category');

    var image = $(this).data('product\_image')

    console.log(name);

    $('#ViewModal').modal('show');

    $('.product\_name').val(name);

    $('.product\_price').val(price);

    $('.product\_category').val(category);

    $('.product\_id').val(id);

    $('img').attr("src", image );

    });

**Login.pug:**

HTML

    HEAD

        TITLE Main

    BODY

        H1 LOGIN/SIGNUP

        if (type == 'success')

            H3 #{message}

        if (type == 'error')

            H3 #{message}

        FORM(action="/home", method="post")

            input(type="text" name="email" placeholder="Email")

            BR

            input(type="password" name="password" placeholder="password")

            BR

            input(type="submit", name="logIn" value='Log In')

        FORM(action="/signup", method="post")

            input(type="submit", name="signUp" value='Sign Up')

**Signup:**

HTML

    HEAD

        TITLE Sign Up

    BODY

        H1 Sign Up

        if (type == 'error')

            H3 #{message}

        FORM(action="/", method="post")

            input(type="text" name="name" placeholder="Name" required)

            BR

            input(type="email" name="email" placeholder="Email" required)

            BR

            input(type="password" name="password" placeholder="Password" required)

            BR

            input(type="password" name="cpassword" placeholder="Confirm Password" required)

            BR

            input(type="submit" name='seller' value='Sign Up as a Seller')

            input(type="submit" name='buyer' value='Sign Up as a Buyer')