```
const express = require('express');
const cors = require('cors');
const mongoose = require('mongoose');
const dotenv = require('dotenv')
const MongoClient = require('mongodb')

dotenv.config();
const app = express();

app.use(cors());
app.use(express.json({limit: "10mb"}))
const PORT = process.env.PORT || 8080

//mongodb connection
mongoose.connect(process.env.MONGO_URL)
   .then(() => console.log("Connected to MongoDB"))
   .catch(err => console.error(err));

   const userSchema = mongoose.Schema({
    firstName: String,
    lastName: String,
    email: {
        type: String,
        required: true,
        unique: true,
    },
    password: String,
    image: String,
   })

   const userModel = mongoose.model('shopifyMart_users', userSchema)

//api
app.get("/", (req, res) => {
    res.send("Hello, World!");
})


// signup
app.post("/signup", async (req, res) => {
    console.log(req.body);

    const { email } = req.body;

    try {
      const existingUser = await userModel.findOne({ email: email });
      if (existingUser) {
        return res.status(400).json({ message: "Email already exists", alert: false });
      }
      const newUser = new userModel(req.body);
      await newUser.save();
      return res.status(201).json({ message: "Registered Successfully", alert: true });
    } catch (err) {
      console.error(err);
      return res.status(500).json({ message: "Something went wrong" });
    }
  })


  //login
  app.post("/login", async (req, res) => {
    console.log(req.body);
    const { email, password } = req.body;

    try {
      const existingUser = await userModel.findOne({ email: email });
      if (!existingUser) {
        return res.status(401).json({ message: "User Not Found", alert: false });
      }

      const isPasswordValid =  (password === existingUser.password);
      if (!isPasswordValid) {
        return res.status(401).json({ message: "Invalid Credentials", alert: false });
      }
      else {
        const dataSend = {
            _id: existingUser._id,
            firstName: existingUser.firstName,
            lastName: existingUser.lastName,
            email: existingUser.email,
            image: existingUser.image,
            alert: true
        }
      return res.status(200).json({ message: "Login Successful", alert: true, data: dataSend });

      }

    } catch (error) {
      console.error(error);
      return res.status(500).json({ message: "Something went wrong", alert: false });
    }
  });

  //product section

  const productSchema = mongoose.Schema({
    name: String,
    description: String,
    price: Number,
    image: String,
    category: String,
  })
  const productModel = mongoose.model('shopifyMart_products', productSchema)

  app.post('/add-product', async (req, res) => {
    console.log(req.body);
    try {
      const newProduct = new productModel(req.body);
      await newProduct.save()
      return res.status(200).json({ message: "Product Added Successfull", alert: true})
    } catch (error) {
      console.error();
      return res.status(500).json({ message: "Something went wrong",  alert: false });
    }
  })


  app.get("/products", async (req, res) => {
      try {
      const products = await productModel.find();
      return res.status(200).json({ message: "Products Fetched Successfully", alert: true, data: products });
      } catch (error) {
        console.error(error);
        return res.status(500).json({ message: "Something went wrong", alert: false });
      }
  })

app.listen(PORT, () => {
    console.log(`Server running on port ${PORT}`);
})
```