**Overview**

This document explains how to deploy a simple Node.js microservices architecture using Docker. The microservices include:

1. **User Service**
2. **Product Service**
3. **Order Service**
4. **Gateway Service**

Each service will be containerized using Docker, and Docker Compose will be used to run and manage all services together.

**Project Structure**

project-root/

├── user-service/

│ ├── app.js

│ └── package.json

├── order-service/

│ ├── app.js

│ └── package.json

├── product-service/

│ ├── app.js

│ └── package.json

├── gateway/

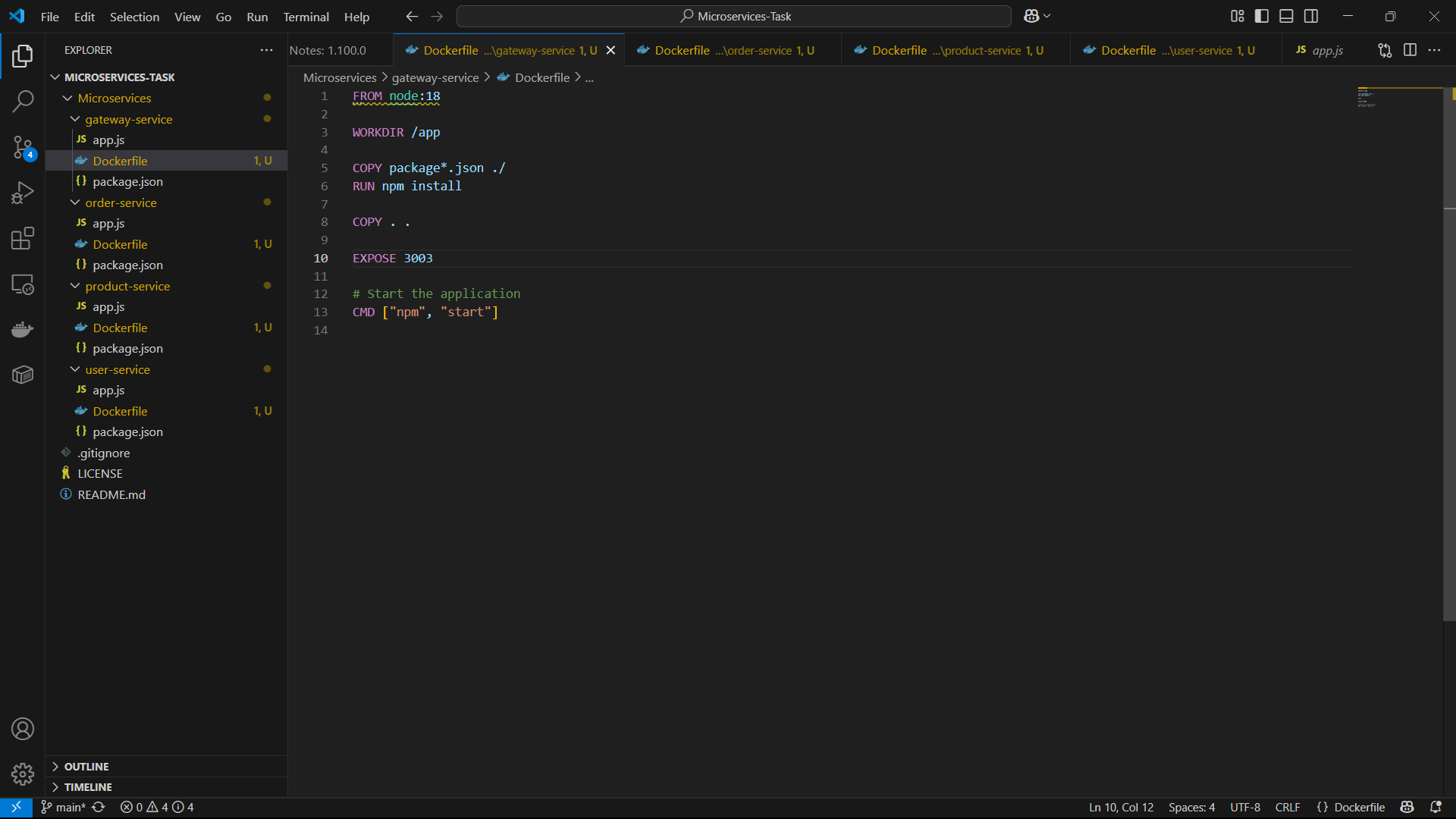
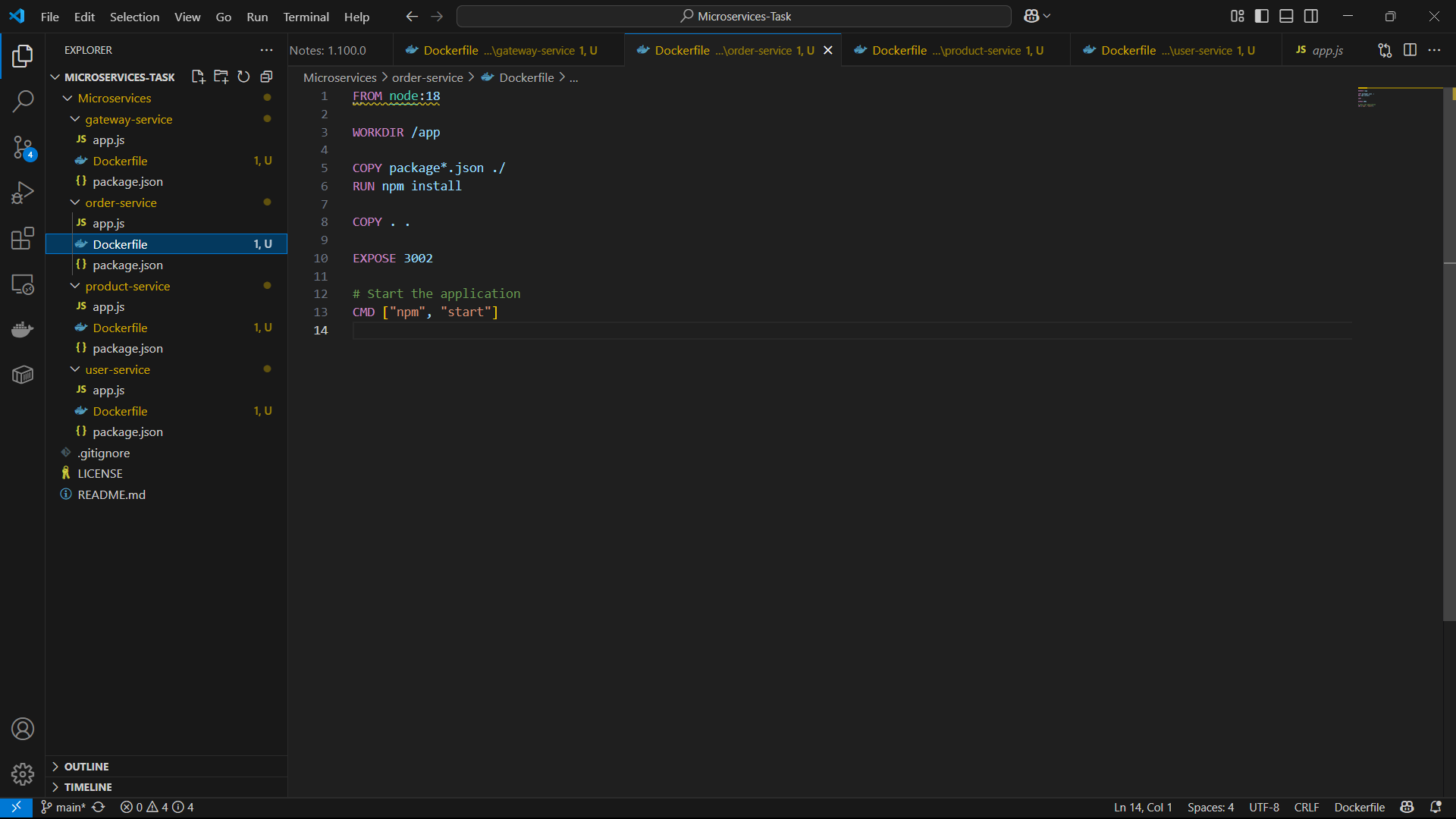
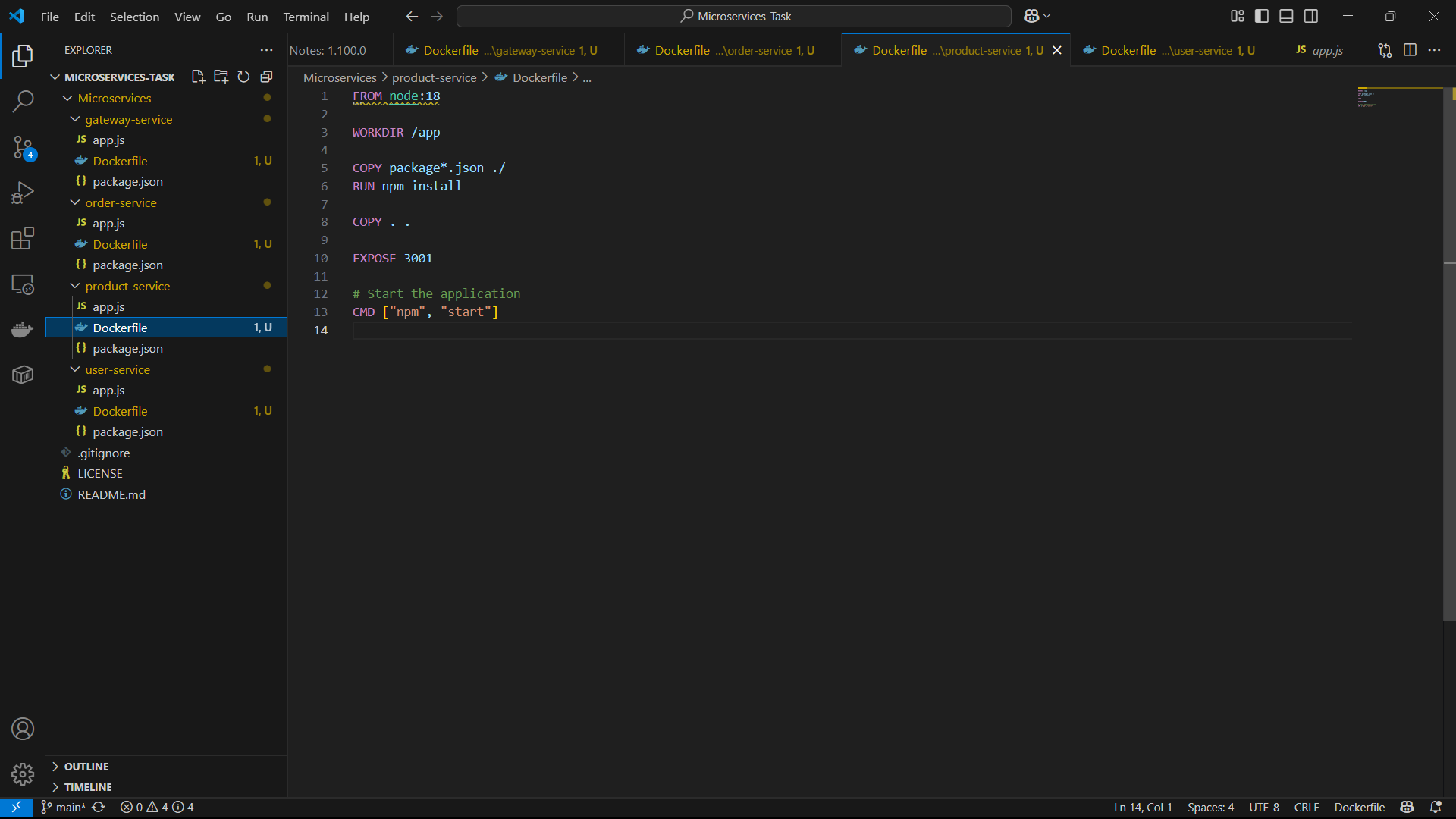
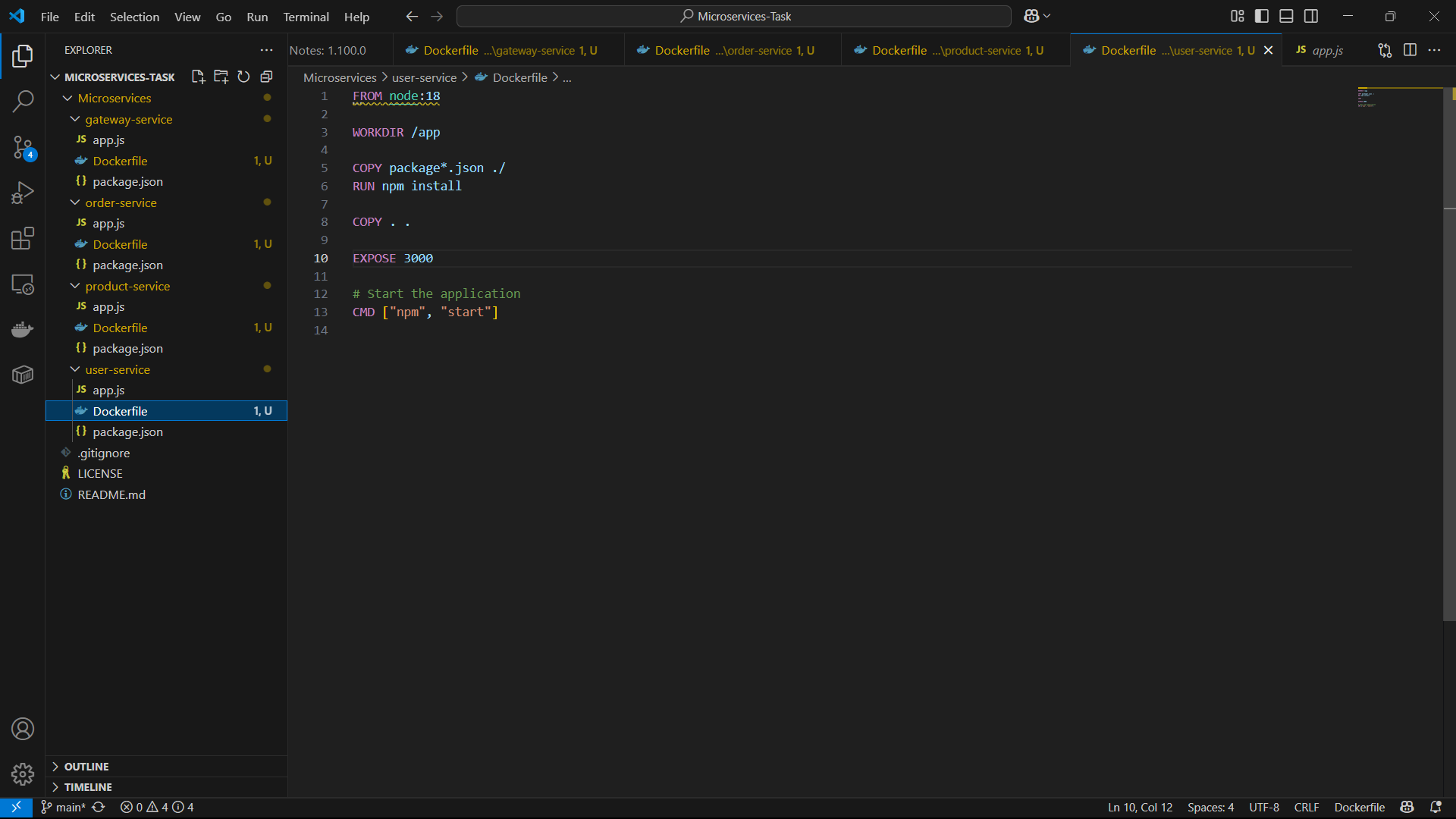
│ ├── app.js

│ └── package.json

├── docker-compose.yml

**Steps to Deploy**

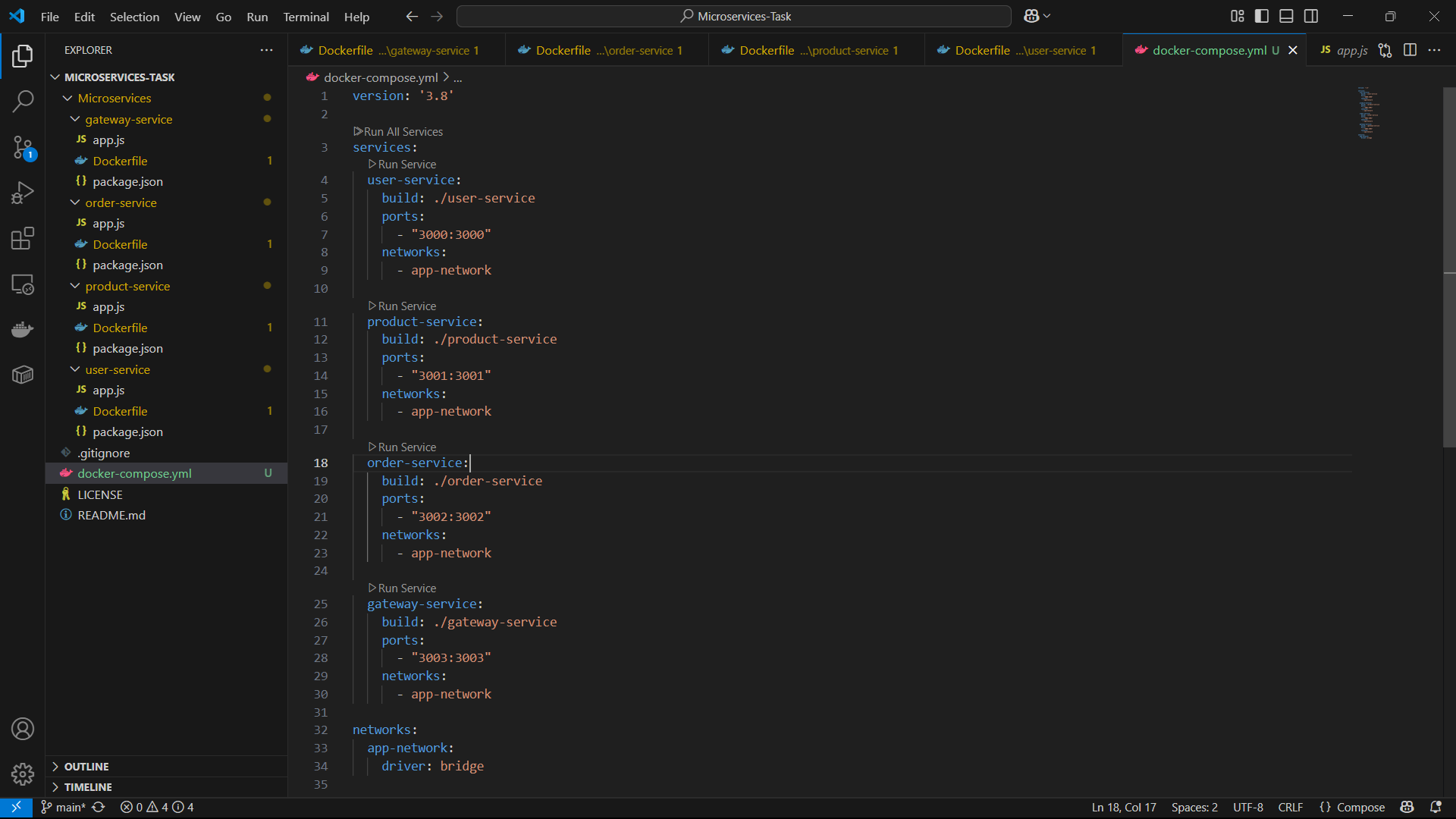
**1. Create Dockerfile for Each Service**

Each service needs a Dockerfile to create its Docker image.   
  
  
  
  
  
  
  
  
  
  


**3. Create docker-compose.yml**

Create a docker-compose.yml file to run all services together. This file defines how each container should run and connect with each other.

Example docker-compose.yml:



**4. Build and Run the Services**

Once all files are set up, run the following command to build and start the services:

docker-compose up --build

This will:

1. Build the Docker images for each service.
2. Start all the services defined in docker-compose.yml.

**5. Verify Services**

Once the containers are running, you can access the services in your browser:

* **Gateway Service**: http://localhost:3000
* **User Service**: http://localhost:3001
* **Order Service**: http://localhost:3002
* **Product Service**: <http://localhost:3003>  
    
    
    
    
    
    
    
    
  