# Ecommerce App

## Backend

* Language: Python
* Framework: Flask, Flask-Injector
* Database: Mongodb
* Cache: redis
* Messaging tool: kafka
* Transaction management: API Gateway
* **Best Practices**
  1. **Use Dependency Injection** (through libraries like **Flask-Injector** or **FastAPI Dependency Injection**) to maintain loosely coupled services.
  2. **Containerize with Docker:** Use Docker for each microservice for isolation and easy deployment.
  3. **Using Kubernetes:** enable Kubernetes in docker desktop, Convert Docker Compose to Kubernetes Manifests.
  4. **CI/CD Pipelines:** Automate testing, deployment, and versioning of each microservice.
  5. **Logging & Monitoring:** Use tools like **Prometheus**, **Grafana**, or **ELK Stack** (Elasticsearch, Logstash, and Kibana) for monitoring and logging.

## Problems:

* Subject must be a string
  1. Related: Authorization token (when calling api after login), when calling create\_access\_token method make sure to use string in identity. Using object instead of string cause this issue.
  2. Solution: Incorrect one identity={"email": user["email"], "role": user["role"]}, correct one (identity=str(user["email"]))
* Need to rebuild every time after any changes. (hot reload)
  1. Solution:
     1. ENV FLASK\_ENV=development in dockerfile
     2. Use volumes in docker-compose.yml
     3. command: flask run --host=0.0.0.0 --port=5002 –reload in docker-compose.yml
     4. use "--reload" in CMD
* Object of type ObjectId is not JSON serializable
  1. When we are storing data in redis it’s causing issue because object id is not properly parsing.
  2. Solution:  user["\_id"] = str(user["\_id"])