WaMa

The Warehouse Management Application

Trevor Mathisen, Viet Nguyen, Hike Yegiyan

CS151: Object Oriented Programming, Spring 2024, Group 6

https://github.com/Yuuji1140/CS151-WaMa

Agenda

- Introduction and Abstract
- Design Pattern (MVC)
- Architecture Overview
- Demo

Introduction

- Application to manage warehouse inventory, orders and shipments
- Customers view products, create orders
- Employees view orders, create shipments
- Current solutions are expensive, complex, bloated with features
- Proposed solution: User-friendly Java GUI application
- Key features:
  - Inventory management
  - Order processing
  - Shipment tracking
  - Customer access
  - Intuitive, up-to-date GUI
  - **Real-time updates**

MVC Design Pattern

- Model
  - Model housed solely in the backend, managed by backend
  - Represented by SQLite Tables (Users, Customers, Products, Orders, Shipments, etc)
  - Maintains state in a thread safe manner
- Viewer
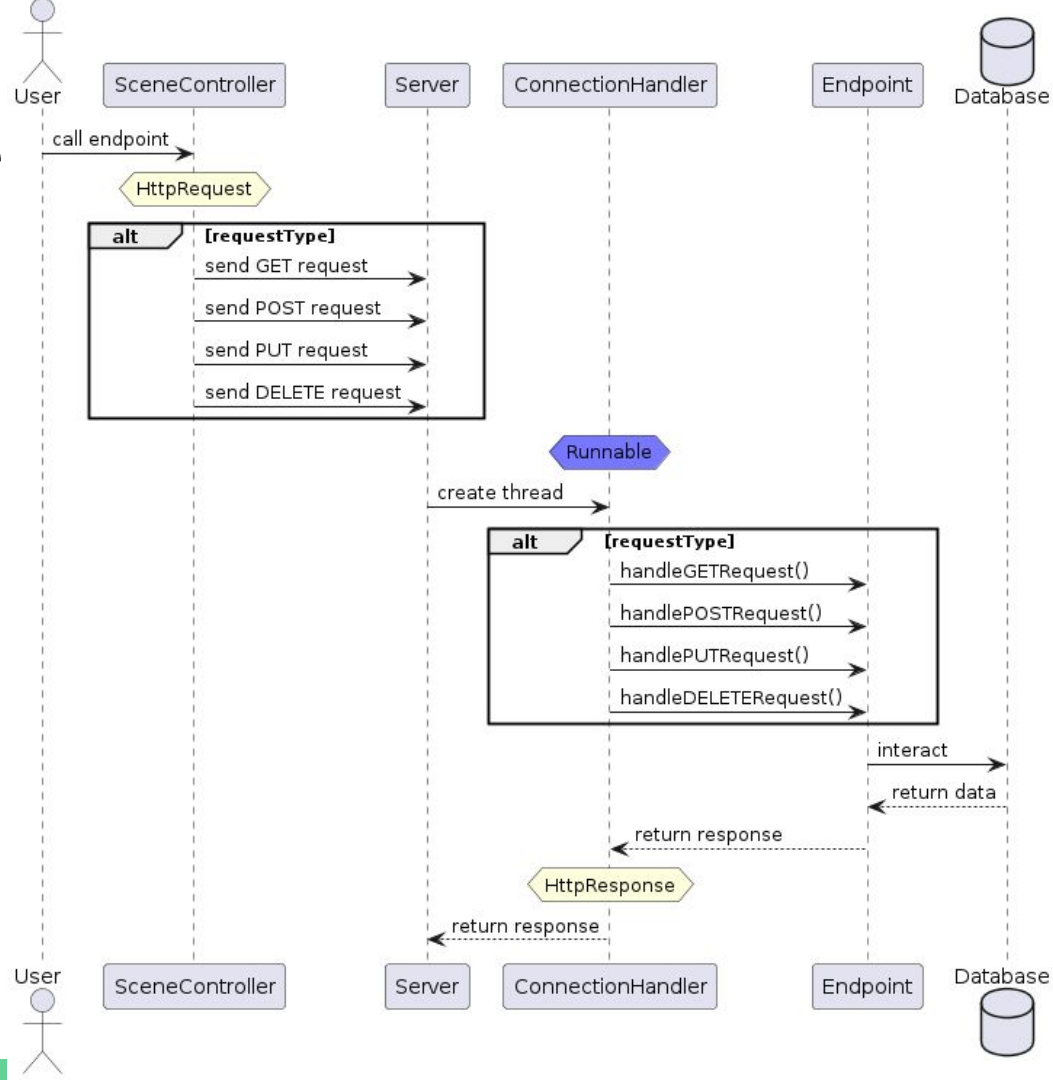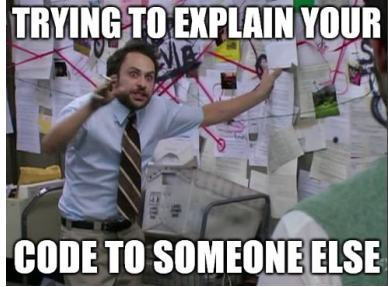 - GUI application each user uses to access the backend
  - Renders predetermined responses to the backend
  - Encapsulates controller logic
  - UpdaterThread maintains current view
- Controller
  - Frontend controller: create drafts, submit requests
  - Backend controller: validate requests, manipulate model

ME MAKING THINGS THREAD SAFE IN JAVA

SYNCHRONIZED

Architecture

TRYING TO EXPLAIN YOUR CODE TO SOMEONE ELSE

User | SceneController | Server | ConnectionHandler | Endpoint | Database

call endpoint

HttpRequest

alt [requestType]
- send GET request
- send POST request
- send PUT request
- send DELETE request

Runnable

create thread

alt [requestType]
- handleGETRequest()
- handlePOSTRequest()
- handlePUTRequest()
- handleDELETERequest()

interact

return data

return response

HttpResponse

return response

Demo