Advanced Programming - Final Project

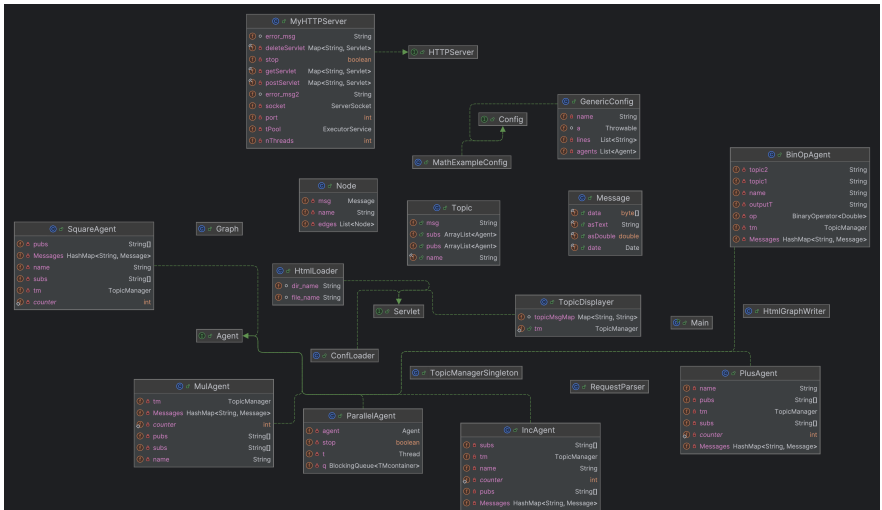
Course Number - 858367701

Ori Ratzon - 213829294
Leeshay Elimelech - 322904756

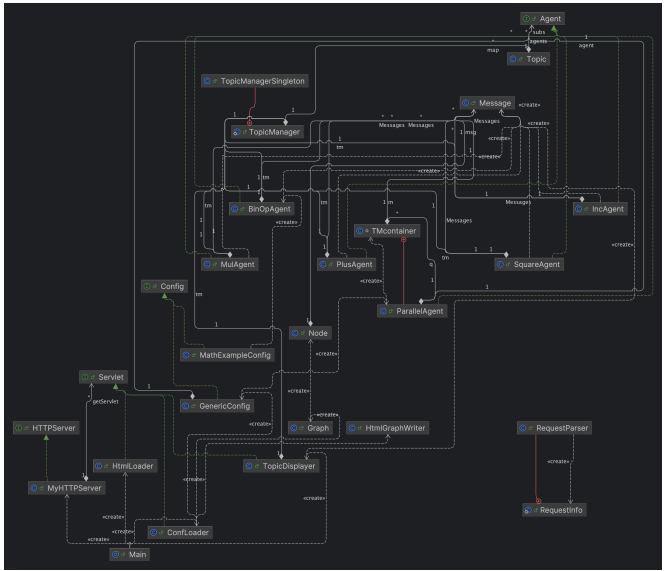
Background

- This project is about simulating a distributed system where each computer runs its own HTTP server.
- The system uses agents and topics, similar to the publish-subscribe model.
- Agents can listen to topics, send messages, and handle data at the same time.
- The project shows how to use communication between programs, work with multiple threads, and build modular code in Java.
- There is also a web interface to set up, view, and interact with the system.

Design


MyHTTPServer
error_msg String
deleteServlet Map<String, Servlet>
stop boolean
getServlet Map<String, Servlet>
postServlet Map<String, Servlet>
error_msg2 String
socket ServerSocket
port int
nPool ExecutorService
nThreads int
HTTPServer
GenericConfig
name String
e Throwable
lines List<String>
agents List<Agent>
Config
MathExampleConfig
BinOpAgent
topic2 String
topic1 String
name String
outputT String
op BinaryOperator<Double>
tm TopicManager
Messages HashMap<String, Message>
Node
msg Message
name String
edges List<Node>
Topic
msg String
subs ArrayList<Agent>
pubs ArrayList<Agent>
name String
Message
data byte[]
asText String
asDouble double
date Date
SquareAgent
pubs String[]
Messages HashMap<String, Message>
name String
subs String[]
tm TopicManager
counter int
Graph
HtmlLoader
dir_name String
file_name String
Servlet
TopicDisplayer
topicMsgMap Map<String, String>
tm TopicManager
Main
HtmlGraphWriter
Agent
ConfLoader
TopicManagerSingleton
RequestParser
PlusAgent
name String
pubs String[]
tm TopicManager
subs String[]
counter int
Messages HashMap<String, Message>
MulAgent
tm TopicManager
Messages HashMap<String, Message>
counter int
pubs String[]
subs String[]
name String
ParallelAgent
agent Agent
stop boolean
t Thread
q BlockingQueue<TMcontainer>
IncAgent
subs String[]
tm TopicManager
name String
counter int
pubs String[]
Messages HashMap<String, Message>

Design


Agent
subs
agents
map
Topic
agent
TopicManagerSingleton
Message
«create»
TopicManager
Messages
msg
BinOpAgent
IncAgent
TMcontainer
MulAgent
PlusAgent
SquareAgent
Config
ParallelAgent
Node
MathExampleConfig
Servlet
getServlet
GenericConfig
HTTPServer
Graph
HtmlGraphWriter
RequestParser
HtmlLoader
TopicDisplayer
MyHTTPServer
RequestInfo
ConfLoader
Main

Live Demo

Extra Features

In addition to the required features, we added some extra functionality to improve the user experience, show better understanding, and demonstrate our design skills. We added:

- `MulAgent` and `SquareAgent` that works similarly to `IncAgent` and `PlusAgent`.
- String agents - `ConcatAgent`, `ReverseAgent`, `UpperCaseAgent` and `LowerCaseAgent`
- A dropdown manu for a layout style.
- A color picker that allows users to choose the colors of topics and agents in the graph.
- Interactive graph view: using the js library "cytoscape", the user can drag and move agents and topics within the displayed graph.
- A feature that lets the user choose how agent names are displayed — either as text or as icons.
- An option to download the current graph, as a standalone html file.
- Dynamic change of font color, when background is dark font is white and vice versa.

What We Learned

- How to build software using interfaces, inheritance, and clear class responsibilities.
- How to create a working HTTP server and handle requests using servlets.
- How to structure a project using layers like model, view, and controller.
- How to manage code properly across multiple files and keep the project organized.

Thanks For Listening