

Computational Graph System

Advanced Programming Final Project

Viktoria Blanko
206649048

17.07.25

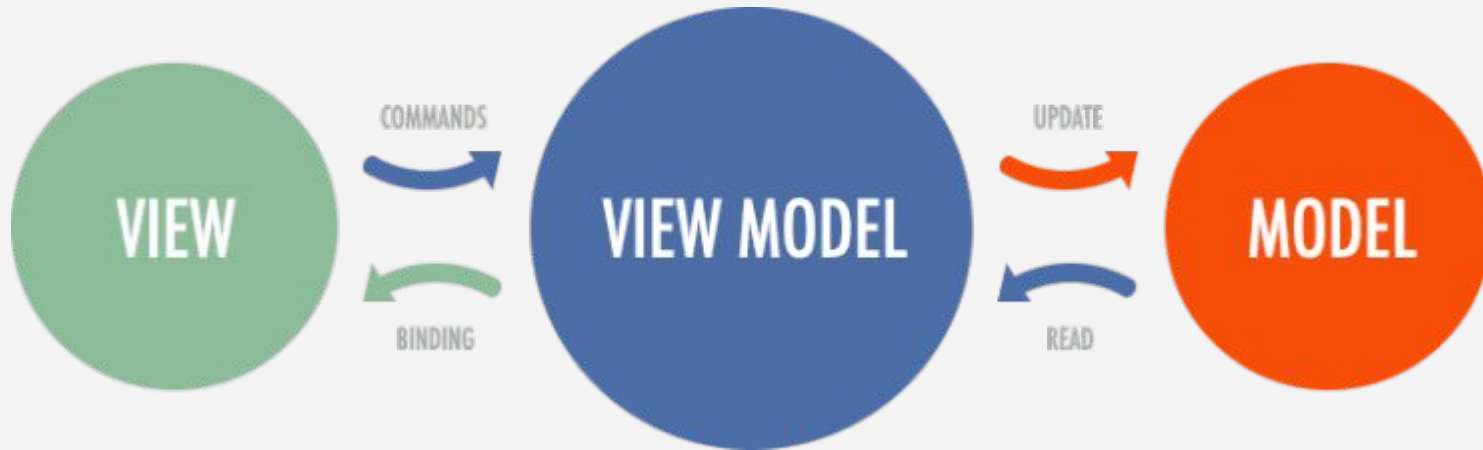
Project Goal

Develop a platform-independent server-client model using a REST-like approach that allows clients to:

- Submit graph configurations and messages
- Receive corresponding visual graphs from the server
- Execute computational graphs remotely through web browsers

Project Architecture

I used a MVVM architecture



Client side



1. Three-panel layout: config upload, graph view, real-time values
2. Uploads .conf files to server via HTTP
3. Visualizes graphs with D3.js (interactive nodes/edges)
4. Updates graph and UI in real time

Choose File simple.conf

DEPLOY

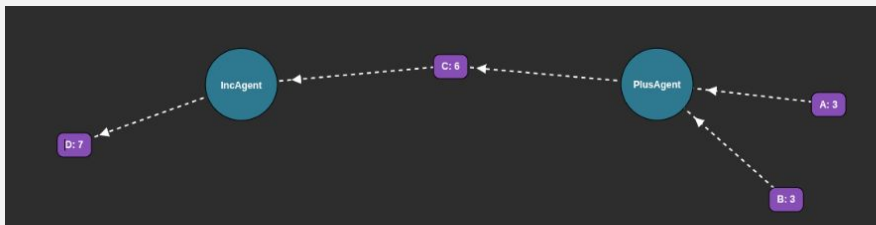
Topic name:

B

Message:

3

SEND



Topic	Value
A	3.00
B	3.00
C	6.00
D	7.00

Server Side

1. Executes mathematical agents (Inc, Plus, Multiply...)
2. Manages pub/sub communication between agents
3. Parses and loads .conf configuration files
4. Generates computational graphs dynamically
5. Listens on port 1234
6. Handles file uploads and serves static web files
7. Responds to real-time topic value requests
8. Stores topic states and agent lifecycles



Frontend special features

01

Drag & Drop Nodes,
Interactive Graph Visualization

02

Drop down list for sending messages
to topics

03

Topics shows the messages they got

04

Handle non numeric messages without
errors

05

Pop up message if files/topics/messages
is empty

06

Clear view of topic values in a table

What i learned?

MVVM Architecture

**Server-Side HTML
Generation**

RESTful Concepts

**Web Application
Development**

**Client-Server
Communication**

**SOLID and GRASP
Principles**

Thank You!