

# CS4262/5462 Course Project Introduction

---

## Building Agentic Applications

TA: Tao Wangcheng

National University of Singapore | 2026 Spring

# Agenda

## PART I Project Requirements

- 01 Requirements I: NLP
- 02 Requirements II: Workflow / Node
- 03 Requirements III: Deliverable & Grading

## PART II Example Implementation

- 04 NUS News ChatBot Overview
- 05 System Architecture
- 06 Workflow Walkthrough
- 07 Execution Engine
- 08 Project Structure
- 09 Environment Setup
- 10 Resources & Q&A

# Requirements I: NLP — AI Generates Agentic Workflow

## Student Tasks

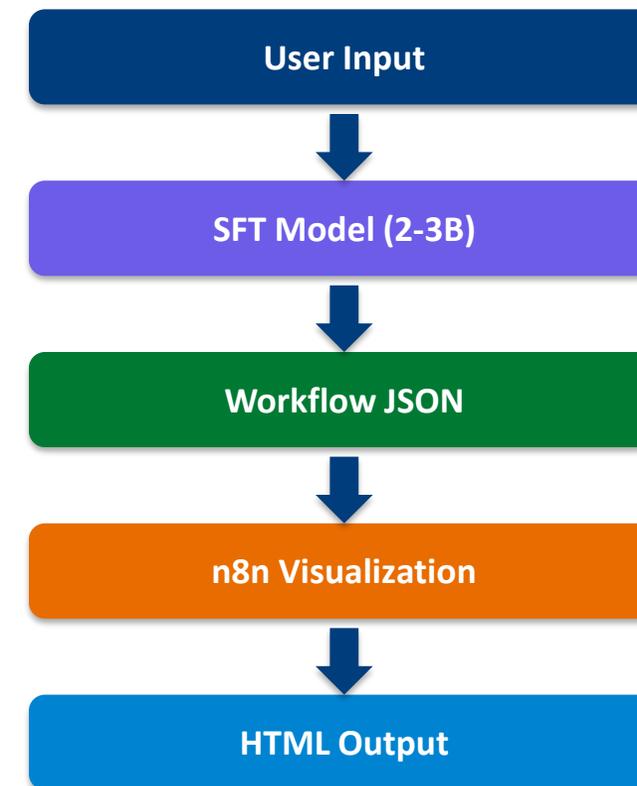
Task	Description	Keywords
a. User App → Workflow	User defines app; AI generates workflow	SFT, JSON, 2-3B
b. Serve the Model	Deploy fine-tuned model, expose API	Model Serving, API
c. Define Workflow IO	Define workflow I/O (e.g. HTML)	IO Schema, HTML

**Bonus: Multi-round Conversation to Update Workflow Graph**

## We Provide

Item	Description
a. Chat Box	n8n workflow JSON visualization + Student API
b. Show Output	Display workflow final / intermediate HTML output

## Overall Flow



# Requirements II: Workflow / Node Implementation

## Student Tasks

Task	Description
a. Node-wise	Implement node functions: LLM, MCP Tool, RAG...
b. Workflow-wise	Dependency mgmt, parallel exec, failure recovery

## Bonus

- Distributed Execution
- Kernel-level Performance Improvements
- Multi-tenancy Support

## We Provide

- Node / Workflow IO Protocol & Standard

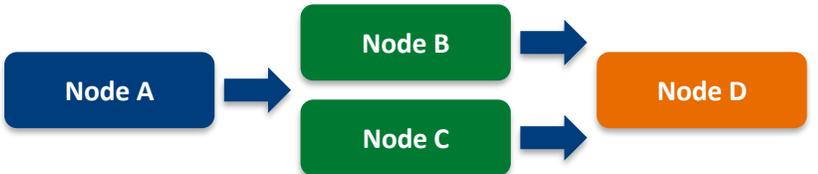
## Node-wise



## Workflow-wise



## Example: Parallel + Dependency



B, C parallel → D waits for both

# Requirements III: Deliverable & Grading

## Final Deliverable

**Student: End-to-End Demo**

### Bonus:

- Bonus A: Standard Workflow Benchmark(Real-world workflow usage, e.g., solving LeetCode problems)
- Bonus B: Update Workflow Graph using RL based on workflow outputs

## Grading Overview

Module	Required	Bonus
I. NLP	a. User App → Workflow (SFT) b. Serve Model & API c. Define Workflow IO	Multi-round conversation to update Graph
II. Workflow/Node	a. Node: LLM, MCP, RAG b. Workflow: Dependency / Parallel / Recovery	Distributed / Kernel-level / Multi-tenancy
III. Deliverable	End-to-End Demo	Standard Benchmark / RL Update

## Important Reminders

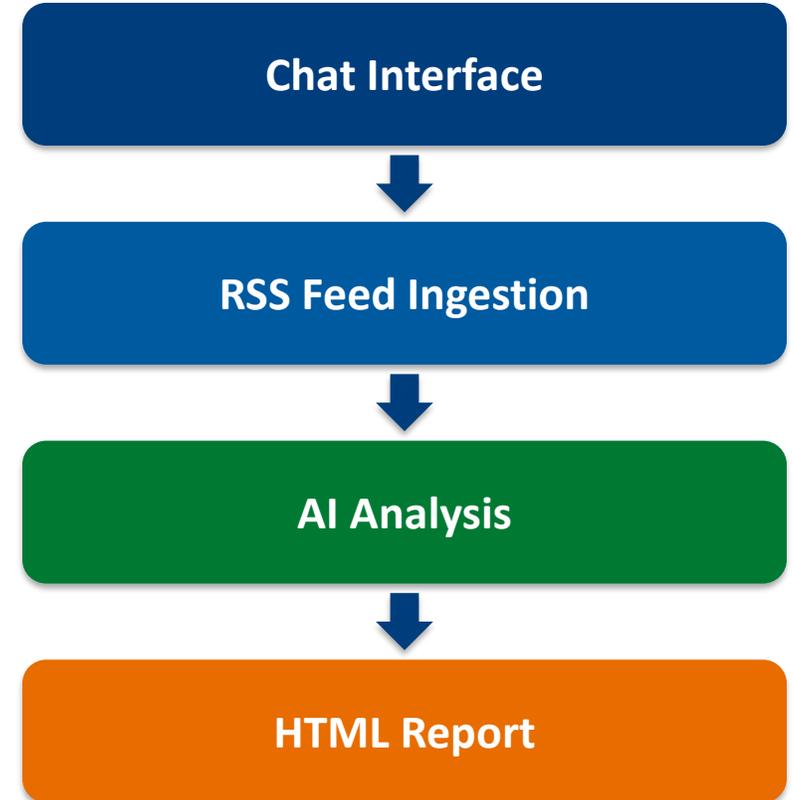
- Codebase is only an example
- You may modify any code (UI, AI, Chat Box)
- You can propose your own features — not limited to the listed bonuses
- Bonus items earn extra credit
- Follow n8n workflow specification, more examples for SFT: <https://n8n.io/workflows/>

# Example: NUS News ChatBot Overview

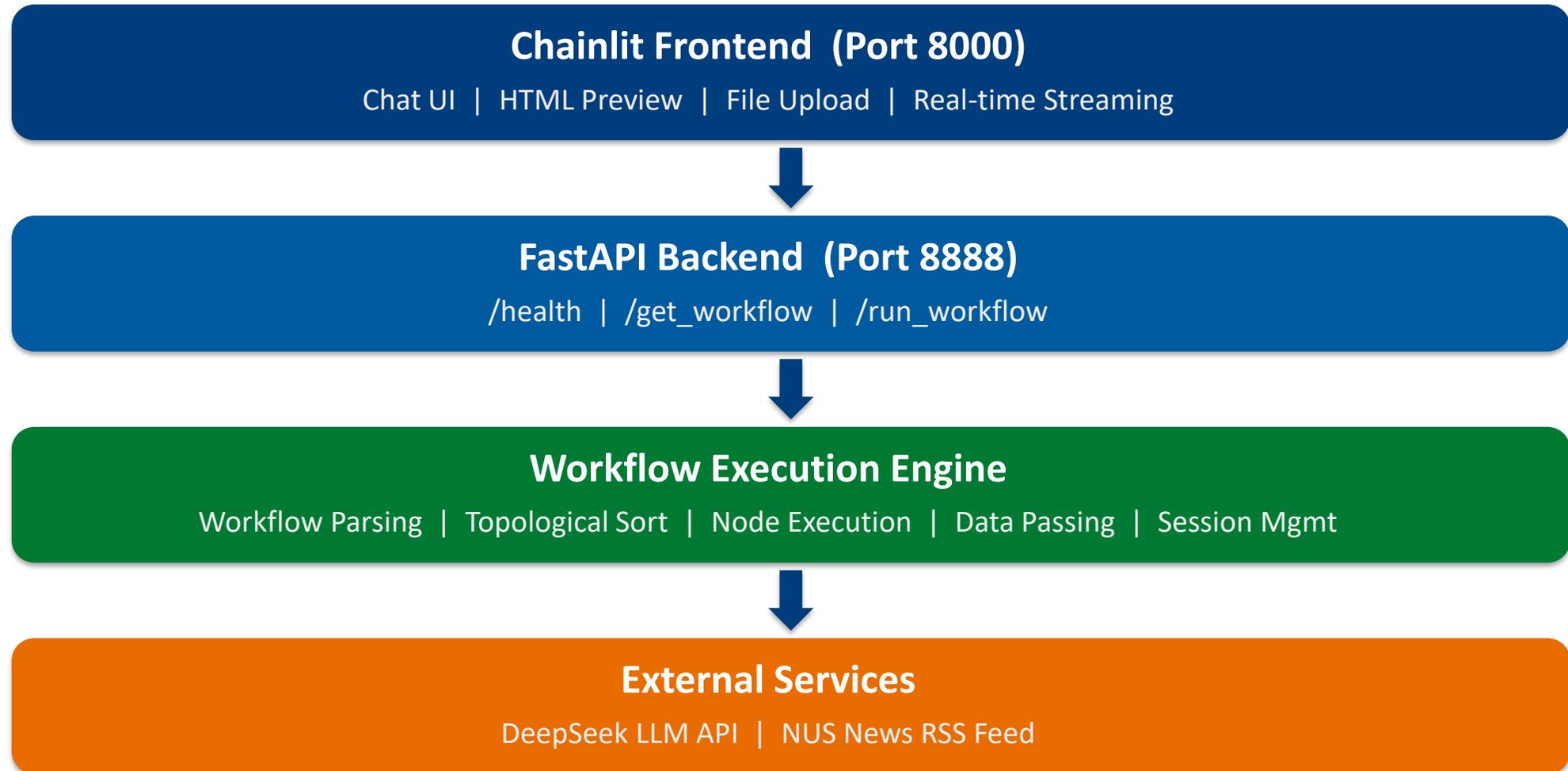
## Key Features

- Users ask questions about NUS news
- System automatically fetches NUS official RSS feed
- AI Agent analyzes and summarizes news content
- Auto-generates HTML-formatted news reports
- Real-time streaming progress display

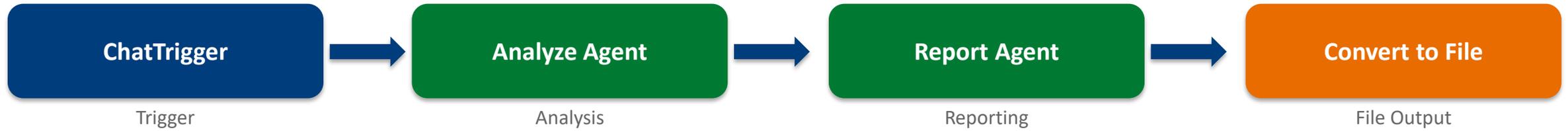
## Pipeline



# Example: System Architecture



# Example: Workflow Walkthrough



## AI Sub-nodes:



## AI Sub-nodes:



## Execution Steps:

1. Trigger: receives user chat message
2. Analyze Agent: uses Code Tool to fetch NUS RSS news, calls DeepSeek for analysis
3. Report Agent: converts analysis into structured HTML report
4. Convert to File: outputs HTML as a downloadable file

# Example: Execution Engine

## Execution Flow

- 1 Load JSON workflow definition file
- 2 Parse Nodes and Connections
- 3 Topological sort for execution order (Kahn's algorithm)
- 4 Execute each main-flow node sequentially
- 5 Pass data between nodes via NodeOutput → NodeInput
- 6 Send real-time notifications after each node completes

## Node Types

Category	Node Type	Description
Main Flow	ChatTrigger	Entry trigger, receives user messages
Main Flow	Agent	AI Agent with tools and memory
Main Flow	ConvertToFile	Converts data to downloadable file
AI Sub-node	MemoryBuffer / ToolCode	Conversation memory / Python code tool
AI Sub-node	DeepSeek LLM / OutputParser	Language model / Structured output

## Connection Types

main	Primary data flow
ai_tool	Tool definition
ai_memory	Conversation memory
ai_languageModel	Language model
ai_outputParser	Output parsing

## Data Passing Model

```
NodeData:
  json_data: Dict[str, Any]
  binary_data: Optional[bytes]
  metadata: Dict[str, Any]

NodeInput: ports[port][source][items]
NodeOutput: ports[port][items]
```

# Example: Project Structure

```
project/
├── environment.yml           # Conda environment config
├── start.sh / stop.sh      # Launch / stop scripts
├── n8n.html                 # Workflow visualization demo
├──
├── chatbot/                # Frontend (Chainlit)
│   ├── app.py              # Main application
│   └── public/elements/    # Custom UI components
├──
└── server/                 # Backend (FastAPI)
    ├── main.py             # API entry point
    ├── n8n_utils.py        # Workflow visualization
    ├── NUS News ChatBot.json # Workflow definition
    └── workflow/           # Workflow execution engine
        ├── models.py       # Data models
        ├── workflow.py     # Parsing (topological sort)
        ├── context.py      # Execution context
        ├── executor.py     # Main executor
        └── nodes/          # Node executors
            ├── agent.py    # Agent (core)
            └── ...         # Other nodes
```

## Key Files

**app.py**

Frontend logic  
Command routing & streaming

**main.py**

Backend API entry  
3 REST endpoints

**executor.py**

Workflow engine  
Node scheduling

**agent.py**

Agent core  
Tool-calling loop

**\*.json**

Workflow definition  
Nodes, connections, params

# Example: Environment Setup & Launch

## Step 1

### Create Conda Environment

```
conda env create -f environment.yml  
conda activate cs4262-5462
```

## Step 2

### Configure API Key(DeepSeek or any other LLM API)

```
echo "DEEPSEEK_API_KEY=your-key" > server/.env
```

## Step 3

### Start Services

```
./start.sh
```

## Step 4

### Stop Services

```
./stop.sh
```

## Access URLs

Service	URL
Chat UI	http://localhost:8000
Backend API	http://localhost:8888
Health Check	http://localhost:8888/health

## Key Dependencies

```
Python 3.12  
chainlit==2.9.6  
fastapi[standard]  
httpx  
openai (DeepSeek API)
```

Troubleshooting: check server/server.log

# Example: Environment Setup & Launch



Readme

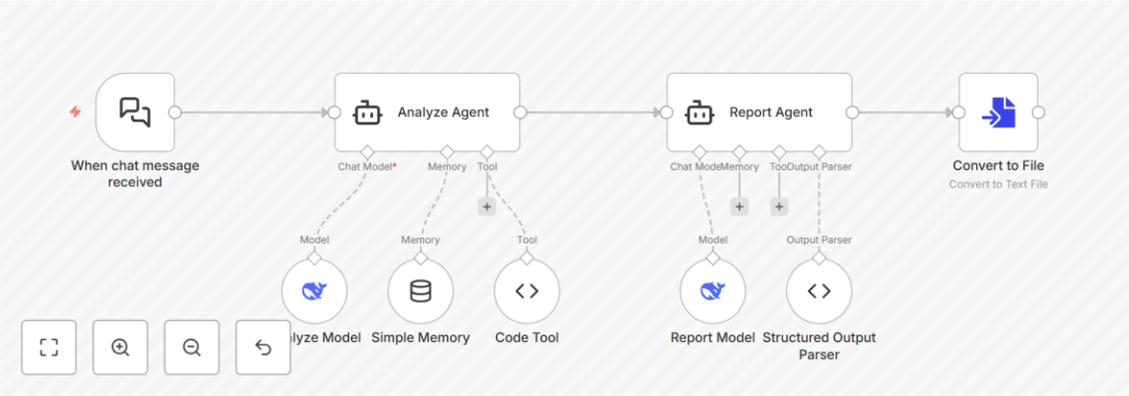
[get\\_workflow](#)  
NUS News ChatBot

Successfully built workflow: **NUS News ChatBot**

HTMLPREVIEW



## ← HtmlPreview



Type your message here...

[get\\_workflow](#) ×



LLMs can make mistakes. Check important info.

# Example: Environment Setup & Launch



Readme



run\_workflow

NUS News from last week

- When chat message received ▾
- Analyze Agent ▾
- Report Agent ▾
- Convert to File ▾
- Workflow executed successfully.

HTMLPREVIEW



Type your message here...



run\_workflow ×



LLMs can make mistakes. Check important info.

← HtmlPreview

## NUS News Report: Week of February 11-18, 2026

Report generated on: February 18, 2026

### 1. Appointment of MAS Distinguished Term Professor

Date: February 16, 2026

**Key Development:** NUS and the Monetary Authority of Singapore (MAS) have jointly appointed **Professor Arvind Krishnamurthy** as the MAS Distinguished Term Professor in Economics and Finance from February 18-28, 2026.

- Professor Krishnamurthy is the John S. Osterweis Professor of Finance at Stanford Graduate School of Business
- Hosted by NUS Business School's Department of Real Estate and MAS's Economic Policy Group
- Will deliver a public lecture on "Dollar Dominance" on February 24, 2026
- This is the 25th MAS Term Professorship awarded since the program began in 2009

### 2. New Book on Social Services Management

Date: February 14, 2026

**Key Development:** A new book titled "Management Issues in Social Services" edited by **Dr S Vasoo**, NUS Social Work Emeritus Professor, was launched.

- Focuses on professional social work's role in delivering cost-efficient, accountable, and impactful social services
- Draws from NUS Social Work's Social Service Leaders Exchange Programme
- Launch event featured Minister Desmond Lee as Guest-of-Honour
- Addresses organizational change, financial management, service evaluation, and stewardship

# Resources & Q&A

## Learning Resources

[Project Docs](#)

README.md in each directory

[Chainlit](#)

[docs.chainlit.io](https://docs.chainlit.io)

[FastAPI](#)

[fastapi.tiangolo.com](https://fastapi.tiangolo.com)

[n8n](#)

[docs.n8n.io](https://docs.n8n.io)

[DeepSeek API](#)

[platform.deepseek.com](https://platform.deepseek.com)

[LangChain](#)

[python.langchain.com](https://python.langchain.com)

[MCP](#)

[modelcontextprotocol.io](https://modelcontextprotocol.io)

# Q & A

Got questions?

**Post your question on Canvas**

[Canvas Discussion](#)

**Thank you!**