Exam MR2 (Option2)

Nom et prénom:
CIN:

Allowed documents:
1- Datacamp Resources
2- List of Figures

Part A : Total: 10 points
1- Insert your proof of progress in the program of '' Building Scalable Agentic Systems''
- Statement of accomplishment (1.5H)

2- Insert your proof of progress in the program of ''Associate AI Engineer for Developers''
- Statement of accomplishment (26H)
- Certificate: AI Engineer for Developers Associate
- Transcript about obtained Score

Part B : Total: 10 points

Title: Agentic RAG System for Tunisian Heritage using Foundation Models and Workflow Orchestration

You are mandated by a Tunisian public institution (e.g. INP, CNCI, regional heritage office) to design and implement an AI assistant that answers questions about Tunisian intangible heritage (poetry, songs, oral stories, rituals, historical events) in Arabic and French. The system must:
- Use foundation models (LLMs and, later, multimodal models) combined with Retrieval Augmented Generation (RAG).
- Be orchestrated using a workflow tool (e.g. n8n style logic: triggers, nodes, branches, sub workflows).
- Respect AI sovereignty constraints (preference for open weight models, local/vectorized storage of Tunisian data, minimal reliance on foreign services for sensitive data).
- Exhibit agentic behavior (the assistant decides when to search, translate, consult an ontology, or refuse).

The exam is split into two phases:

Phase 1 – In class Design (No Internet)

All answers are on paper. You must give concrete, detailed visuals and algorithms : block diagrams, sub block diagrams, workflows, sub workflows, and pseudo code.

1. System level and sub block diagrams (1 page)

1.1 Global architecture diagram (0.5 page)
Draw a top level block diagram of the Tunisian heritage assistant.
For each block, annotate:
- Its main function (1 short phrase).
- Its main inputs and outputs (e.g. “raw PDFs → cleaned text chunks”).

1.2 Sub block diagram of one core component (0.5 page)
Choose one of the key components and:
- Decompose it into few sub blocks
- Draw a sub block diagram showing arrows between sub blocks and label the data that flows along each arrow.
- Add 1–2 sentences explaining how this component works internally, consistent with your top level diagram.

2. Workflows and sub workflows (1 page)

Here you describe the behavior of the system as workflows, similar to a node based orchestration tool.

2.1 Ingestion / indexing main workflow + sub workflow (0.5 page)
a. Draw an ingestion workflow that covers multiple steps as boxes or nodes with arrows.
b. Then, for a step (example: Preprocessing) , draw a more detailed sub workflow that includes conditional branches.

2.2 Query / agent main workflow + sub workflow (05 page)
a. Draw the main query workflow with at least these elements:
b. Then, draw a sub workflow for the tool selection step, showing at least three different branches, .. . Label each branch with the condition and the chosen tools.

3. Agent decision pseudo code and evaluation hooks (1 page)
3.1 Pseudo code for the agent decision loop (0.5 page)
Write language agnostic pseudo code that implements the core decision logic consistent with your diagrams, showing:
Use clear `IF / ELSE IF / ELSE` or `MATCH` structures, and show how the agent may refuse or request clarification when sovereignty or ethics checks fail.

3.2 Evaluation hook points (0.5 page)

Indicate explicitly in your pseudo code or on your diagrams where evaluation will be applied in Phase 2, for example:
- After `Generate_LLM_Response`: measure heritage answer quality and hallucination risk.
- After tool selection: measure percentage of queries requiring foreign services.
- After logging: enable future analysis of bias, coverage, and failure modes.

Phase 2 – Out of class Implementation & Evaluation (optional)

In this phase you can use Internet, workflow tools (e.g. n8n), APIs, and any allowed open source components. You must implement and evaluate the design from Phase 1, with minimal justified adaptations.

4. Implementation and demonstration

Using a workflow/orchestration tool of your choice (e.g. n8n style system):

- Implement the ingestion/indexing workflow and the query/agent workflow that correspond to your Phase 1 diagrams (adaptations must be explained briefly).
- Provide evidence of execution (screenshots of workflows, log excerpts, and at least 3 example queries and outputs that show:
  - a purely heritage RAG answer,
  - a case requiring translation,
  - a case that should be refused or redirected).

Your implementation can be minimal but must be coherent with your initial design.

5. Evaluation and critical reflection

Prepare a concise report (1–2 pages) that includes:

5.1 Test and evaluation results
- At least 5 test queries (including some from Phase 1) that cover:
  - Different languages (Arabic/French).
  - Different content types (poetry, rituals, historical events).
  - At least one “difficult” or ambiguous case.
- For each query, present a small table or structured list indicating: input, system answer, and scores according to the metrics you proposed (e.g. relevance, factuality, cultural adequacy, use of sovereign vs foreign tools).

5.2 Critical reflection and improvement
- In a short section, discuss:
  - One strength and one limitation of your architecture and implementation (technical or cultural/ethical).

- One concrete improvement you would make if you had more time (e.g. better ontology integration, improved evaluation pipelines, additional safety filters, stronger sovereignty guarantees).