AMD AI Sprint Hackathon 2025

Track: AAIPL – Q-agent vs A-agent
 Team Folder: AAIPL_129_212_191_5

Github link : https://github.com/Devidasblr/AMD-Hackathon

👥 Team Composition

- **Q-agent (Question Generator)**

    - **Model Used**: Qwen/Qwen1.5-4B

    - **Purpose**: Generate logic-based MCQs (multiple-choice questions) with options and correct answers.

- **A-agent (Answer Generator)**

    - **Model Used**: Qwen/Qwen1.5-4B

    - **Purpose**: Given a question, choose the correct option and explain the reasoning.

✅ Design Highlights

🧩 Question Agent (`q_agent.py`)

- **Prompt Engineering**:

    - Used system prompt to act as a logic-based MCQ creator.

    - Ensured the output is structured with question, options, and answer.

- **Inference Pipeline**:

    - Supports --num_questions as CLI argument.

    - Controlled max_tokens, temperature, etc., via qgen.yaml.

- ○ Filters invalid formats before saving to `questions.json`.

🧪 Answer Agent (`a_agent.py`)

- **Prompting Strategy**:

  - ○ Model instructed to act as a logical MCQ solver.

  - ○ Output is expected in JSON with `answer` and `explanation`.

- **Processing**:

  - ○ Reads from `questions.json`.

  - ○ Stores responses in `answers.json`.

📁 Folder Structure

```
pgsql
CopyEdit
AAIPL_129_212_191_5/
├── q_agent.py
├── a_agent.py
├── qgen.yaml
├── agen.yaml
├── ckpt/
│   └── qwen1.5-4b/
├── questions.json
├── answers.json
└── solution.pdf
```

📏 Compliance Checklist

- ● ✅ No RAG or retrieval-based modules

- ✅ Only Qwen1.5-4B used for both agents

⚠️ Notes

- Models and tokenizer loaded via Hugging Face Hub.

- Manual login/token used to authenticate access.

- All YAML-configured parameters handled during inference.