

Task Deployment: Bhavesh

Task Set: Dual Micro Modules (Deadline: 9th Nov)

Module 1 — “Shruti Parser” (LangChain Micro Agent)

Build a lightweight LangChain-based text analysis microservice that:

- Accepts any text input (API or CLI).
- Classifies it into one of four intents: command, query, teaching, data.
- Extracts key instructional fragments or “action phrases.”
- Returns structured JSON output.

Deliverables:

- app.py or main.ipynb with core logic.
- Three endpoints or methods: /analyze, /extract, /summarize.
- Example inputs + outputs in a test_data folder.

Learning Focus:

Prompt templates, parsing chains, JSON structure, modular API logic.

Module 2 — “Mini LoRA Benchmark Kit” (ModelOps Task)

Create a local fine-tuning + benchmarking setup using Hugging Face or Ollama.

- Choose a small base model (phi-1.5, mistral, or distilbert-base).
- Apply lightweight LoRA tuning on a mini dataset (10–20 samples).
- Benchmark: latency, token response speed, and quality vs base.

Deliverables:

- Jupyter notebook with LoRA fine-tuning + inference.
- Results JSON or CSV with timing metrics.
- Short markdown report (REPORT.md) comparing results.

Learning Focus:

Fine-tuning pipeline, local model deployment, comparative evaluation.

Timeline (Nov 4–9)

- Nov 4–5: Shruti Parser design + testing.
- Nov 6–7: LoRA setup + mini fine-tune.
- Nov 8: Combined documentation + cleanup.
- Nov 9: Final submission via GitHub + Loom walkthrough.

Submission Package

- Repo name: Shruti-Parser-LoRA-Benchmark
- Include:
 - /shruti_parser
 - /lora_benchmark
 - /docs (with Loom link + markdown)