COMPUTATIONAL INTELLIGENCE | MRCET

(CSE – ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

Email: mrcetsocse4@mrcet.ac.in

IV Year B. Tech-I Semester Mini Project Summary Sheet

Project Title:	**Memora: Document-Aware AI for Goal-Centric Personal Assistance and Customized Task Scheduling**				
Project Code:		**Batch Size:**	03	**Batch:**	2022 – 26
Domain / Area:	Generative AI	**SDG Mapping**	Industry, Innovation and Infrastructure		

Abstract:

Memora is a local AI-powered scheduling assistant designed to redefine personal productivity by combining natural language interaction with intelligent calendar management. Unlike cloud-based tools, it ensures complete data privacy by running entirely on-device, leveraging **FastAPI**, **Next.js**, and **Ollama with Qwen 2.5 7B** for local LLM inference. The system introduces a conversational interface capable of handling complex scheduling requests, detecting conflicts, and integrating with personal documents through **retrieval-augmented generation (RAG)**. With real-time synchronization across chat and calendar views, semantic search, and bulk task operations, Memora delivers an intuitive yet privacy-first productivity solution while laying the foundation for future collaborative and multi-modal AI assistants.

Technical (S/w & H/w) Specifications

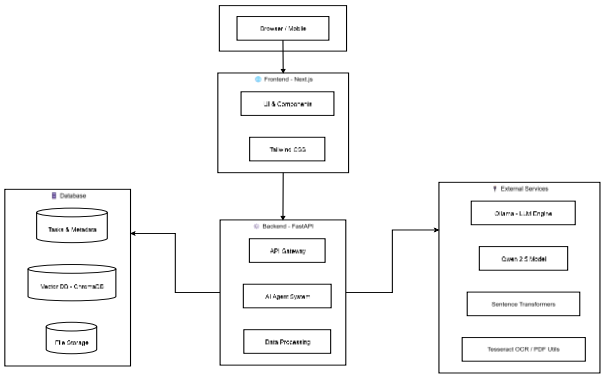
Software Requirements
- Python 3.12
- Node.js 18
- Ollama (qwen2.5:7b)
- ChromaDB
- Poppler, Tesseract

Hardware Requirements:
- 16GB RAM
- Ryzen 7 5800H
- RTX 3050
- 10GB free space

Module(s) Specifications

Module 1: UI — Chat, calendar, documents.
Module 2: Chat — AI conversation I/O.
Module 3: Agent — Intent, context, conflict resolution, planning.
Module 4: Vector Memory — ChromaDB embeddings + document OCR/RAG.
Module 5: Task Store/API — JSON tasks CRUD + FastAPI endpoints

Architecture Diagram



Methodology

1. Chat → intent → task ops → response → UI sync.
2. LangGraph agent: context + conflict handling.
3. Qwen 2.5 via local Ollama (low temp, hybrid CPU/GPU).
4. RAG: extract → OCR → chunk → embed → Chroma search.
5. Storage: JSON tasks; SQLite-backed Chroma vectors.
6. Endpoints: /chat, /tasks, /tasks/today, /tasks/stats, /health.

Existing System

1. Manual entries, forms.
2. No AI intent/conflict handling.
3. Cloud-tied, privacy risk.
4. Fragmented chat/calendar/docs.

Proposed System

1. ANL chat scheduling with AI.
2. Auto conflict detection/resolution.
3. Local-first, offline, private.
4. Unified chat + calendar + docs.

Guide Details

Talluri Haribabu
Assistant professor in Department of CSM
MRCET Campus
UGC Autonomous Institution
-Govt. of India
Hyderabad

Batch Members Details

Durishetty Anirudh
22N31A6648

Bonjuri Raghuvardhan
22N31A6631

Arelli Karthikeya
22N31A6611