

★ ELIAS – Day 12 (Saturday) Official Work Plan

(8 PM – 10 PM Work Session)

Theme: Deep Integration Weekend – Making the System “One Piece”

This is one of the MOST important work days.

Today, each module begins interacting realistically with others.

✂ Team Roles for Day 12

Member	Role	Main Focus
M S	Lead & Integrator	Combine NLP + Plugins + UI + Logs
S S	NLP	Expand hybrid model + entity improvements
A K	Vision	Implement actual face-auth logic
R A	Plugins	Add 2 new plugins + refine previous ones
M A	QA & Documentation	End-week documentation + test suite

■ 1. M S — Lead & Integrator

Today's Focus: Create the “Core Pipeline”

You will create **one unified file** that acts as the official backend API:

src/core_engine.py

This file integrates:

- NLP (parse_intent)
- Controller logic
- Plugins dispatcher
- Vision Auth check (optional)
- Logging
- Return structured responses for Streamlit UI

Tasks

- ✓ Build this official function:

```
def elias_engine(user_input):
```

```
    return {  
        "intent": ...,  
        "entities": ...,  
        "message": ...,  
        "ok": True/False,  
        "confidence": ...,  
    }
```

✓ Connect this to Streamlit:

```
result = elias_engine(text_input)
```

```
st.write(result["message"])
```

✓ Add central error handling

✓ Add debug mode toggle

✓ Refactor messy code from app.py into modules

Expected Output

- A unified “brain” (core_engine.py)
- UI connected to engine successfully
- System behaves like a real assistant

■ 2. S S — NLP Developer

Today's Focus: Hybrid NLP Upgrade

✓ Add semantic keyword groups:

- greetings
- ask_time
- ask_date
- clarification
- exit

- help

✓ Add **word vector similarity lite** (no ML libs) using:

common words → synonyms list

✓ Improve entity extraction:

- Extract multi-word names
- Extract numbers
- Extract verbs
- Extract app names (chrome, notepad, etc.)

✓ Add 1 new intent:

play_music_local

Expected Output

- NLP more accurate + flexible
- New intents recognized
- Entity extraction upgraded

3. A K — Vision Engineer

Today's Focus: Real Authentication Logic

✓ Capture a user's face embedding (Mediapipe or dlib fallback)

✓ Save as:

data/auth/user_face.json

✓ Add actual matching function:

compare_embeddings(v1, v2) -> confidence score

✓ Integrate with M S's core_engine:

- Sensitive actions require authentication
- If face mismatch → return auth error

✓ Add “face_not_found” graceful output

Expected Output

- Real face auth skeleton working
- Controller can call authentication from engine

4. R A — Plugin Developer

Today's Focus: Two New Plugins

Select 2 from list below:

Option A — System Info Plugin

- CPU %
- Memory %
- Disk %
- Battery %
- OS info

Option B — Reminder Plugin

- Save reminder to data/reminders.txt

Option C — Calculator Plugin

- Evaluate arithmetic safely

Option D — Weather Plugin (API-free)

- Parse user query
- Return structured weather placeholder data

Option E — File Actions Plugin

- Create file
- Delete file
- Rename file

✓ Wrap each plugin in optimized structure.

✓ Test plugins via core_engine NOT directly.

Expected Output

- Two new plugins functional

- Routed through the system correctly

5. M A — Documentation & QA

Today is Documentation HEAVY

✓ Write “**Week 2 Summary Document**”

✓ Update folder-level README

✓ Create a “Testing Sheet” in markdown:

- Test 10 NLP inputs
- Test 10 plugin commands
- Test 5 vision checks
- Test 5 UI interactions

✓ Prepare a “Mid-Project Report”

✓ Update diagrams (architecture + flow)

End-of-Day (Saturday) Expected Result

By end of Day 12:

✓ core_engine.py created

✓ UI ↔ Engine working smoothly

✓ NLP advanced

✓ Vision auth semi-functional

✓ 2 new plugins created

✓ Mid-project documentation written

This is the moment where:

 *ELIAS becomes a functional early assistant.*