

ELIAS – Day 13 (Sunday) Official Work Plan

(Core Development Sprint — FINAL PHASE BEGINS)

(Work Session: Flexible — 3 to 4 hours total)

Team Roles (Reference Only)

| Member | Role | Responsibility |
|--------|---------------------------|--|
| M S | Project Lead & Integrator | Merges, optimizes, keeps code structure stable |
| S S | NLP Lead | Extends powerful hybrid NLP with more intents |
| A K | Vision Engineer | Builds face-recognition authentication |
| R A | Plugin Developer | Develops useful action modules |
| M A | Documentation & QA | Ensures overall documentation + test coverage |

Day 13 — Focus Theme:

“Connect Vision + NLP + Plugins into a Full System”

Yesterday (Day 12) we completed:

- ✓ UI skeleton
- ✓ Logging
- ✓ Optimized NLP
- ✓ Create Note plugin
- ✓ YouTube plugin
- ✓ Website plugin working
- ✓ All modules talking to app.py

So Day 13 is about **making the system cohesive & adding depth.**

1. M S — Project Integrator (Major Task Today)

Task 1 — Merge Vision into workflow (non-blocking)

Integrate vision auth into app.py in optional mode:

- Before executing **sensitive intents**, call detect_face()
- Add a config: REQUIRE_AUTH = False/True

- Mark future intents like "delete_note" or "open_settings" as sensitive.

Output:

Updated app.py with conditional vision-auth sections.

 **Task 2 — Create “Intent → Category” map**

Make a small file:

src/intent_categories.py

```
INTENT_CATEGORIES = {
```

```
    "open_website": "web",
    "search_youtube": "media",
    "create_note": "notes",
    "read_note": "notes",
}
```

This will allow the UI to:

- Group commands
- Show usage tips
- Build analytics (later)

Output:

New module + imported in UI.

 **Task 3 — Clean folder structure**

Ensure:

ELIAS/

 src/

 nlp/

 plugins/

 vision/

 ui/

logs/

docs/

Output:

Final folder structure ready for Monday (Day 14 polish).

2. S S — NLP Developer (Strong but manageable day)

Task 1 — Add 3 new intents

Suggested:

- "read_note" (entity: title)
- "open_youtube_video" (entity: url or id)
- "search_web" (entity: query)

Task 2 — Add fallbacks

If unknown intent:

- Check if input looks like a URL
- Check if input looks like “search ____”

Add fallback rules in parse_intent():

if looks_like_url:

```
return {"intent": "open_website", ...}
```

Task 3 — Improve accuracy

Expand keyword tokens for prefilter → faster + more accurate.

Output:

Updated hybrid NLP module + tested with app.py.

3. A K — Vision Engineer (Today is BIG for AK)

Task 1 — Build facial_encoding loader

Create:

vision/face_recognition_core.py

Loads:

- known face encodings (from image in /assets)
- use face_recognition library

Task 2 — Implement detect_face() final version

Steps:

1. Capture webcam frame
2. Detect face
3. Compare with known encoding
4. Return True/False

Output:

Working face-authentication backbone.

4. RA — Plugin Developer

Task 1 — Implement read_note plugin

File: plugins/read_note.py

- Locate note by title or list notes
- Return its text
- Safely open correct file

Task 2 — Add basic calculators plugin

plugins/calc.py

Uses:

- regex to extract numbers
- eval for simple expressions
- safe sandboxing

Task 3 — Begin “play_music” (local audio)

Optional:

- Play .mp3 from /assets/music/

Output:

New functional plugins.

5. M A — Documentation & QA

Task 1 — Write Day 13 log

Include:

- NLP extension notes
- Vision progress summary
- New plugins
- Folder structure update
- Screenshots of working features (optional)

Task 2 — Update Documentation Index

Add sections:

- "Vision System"
- "New Intents"
- "Notes Plugin"
- "System Integration Flow"

Task 3 — Create QA checklist

Checklist for tomorrow testing:

- All intents respond
- All plugins run
- Logging works
- UI shows updates
- No missing imports

Day 13 End-of-Day Goals (Entire Team)

By tonight:

- ✓ Vision system is functional
- ✓ NLP supports more intents + fallbacks
- ✓ New plugins: read_note, calc, play_music
- ✓ App.py integrated & fully modular
- ✓ Folder structure cleaned
- ✓ Documentation updated

Meaning:

- 👉 The CORE SYSTEM becomes **95% complete**
- 👉 Day 14–16 will only be about:

- UI polish
- Testing
- Performance
- Documentation
- Rehearsal

Perfect pacing.