```python
from fastapi import FastAPI
from pydantic import BaseModel
from typing import List
import threading
import uvicorn

# Import modules
from nlp_extractor import extract_drug_info
from drug_interaction import get_drug_interactions
from dosage_recommendation import get_age_specific_dosage
from alternative_medication import suggest_alternatives

app = FastAPI()

class PrescriptionRequest(BaseModel):
    prescription_text: str
    patient_age: int

@app.post("/verify")
def verify_prescription(request: PrescriptionRequest):
    text = request.prescription_text
    age = request.patient_age

    # Extract drugs
    extracted = extract_drug_info(text)
    drugs = extracted["drugs"]

    if not drugs:
        return {
            "extracted_drugs": [],
            "interactions": [],
            "recommended_dosages": {},
            "alternatives": {},
            "note": "No drugs detected. Check input or try clearer text."
        }

    # Check interactions
    interactions = get_drug_interactions(drugs)

    # Dosage
    dosages = {}
    for drug in drugs:
        dosages[drug] = get_age_specific_dosage(drug, age)
```

```python
        # Alternatives if interactions exist
        alternatives = {}
        if interactions:
            for drug in drugs:
                alternatives[drug] = suggest_alternatives(drug)

        return {
            "extracted_drugs": drugs,
            "interactions": interactions,
            "recommended_dosages": dosages,
            "alternatives": alternatives
        }

def run_fastapi():
    uvicorn.run(app, host="0.0.0.0", port=8000)
```