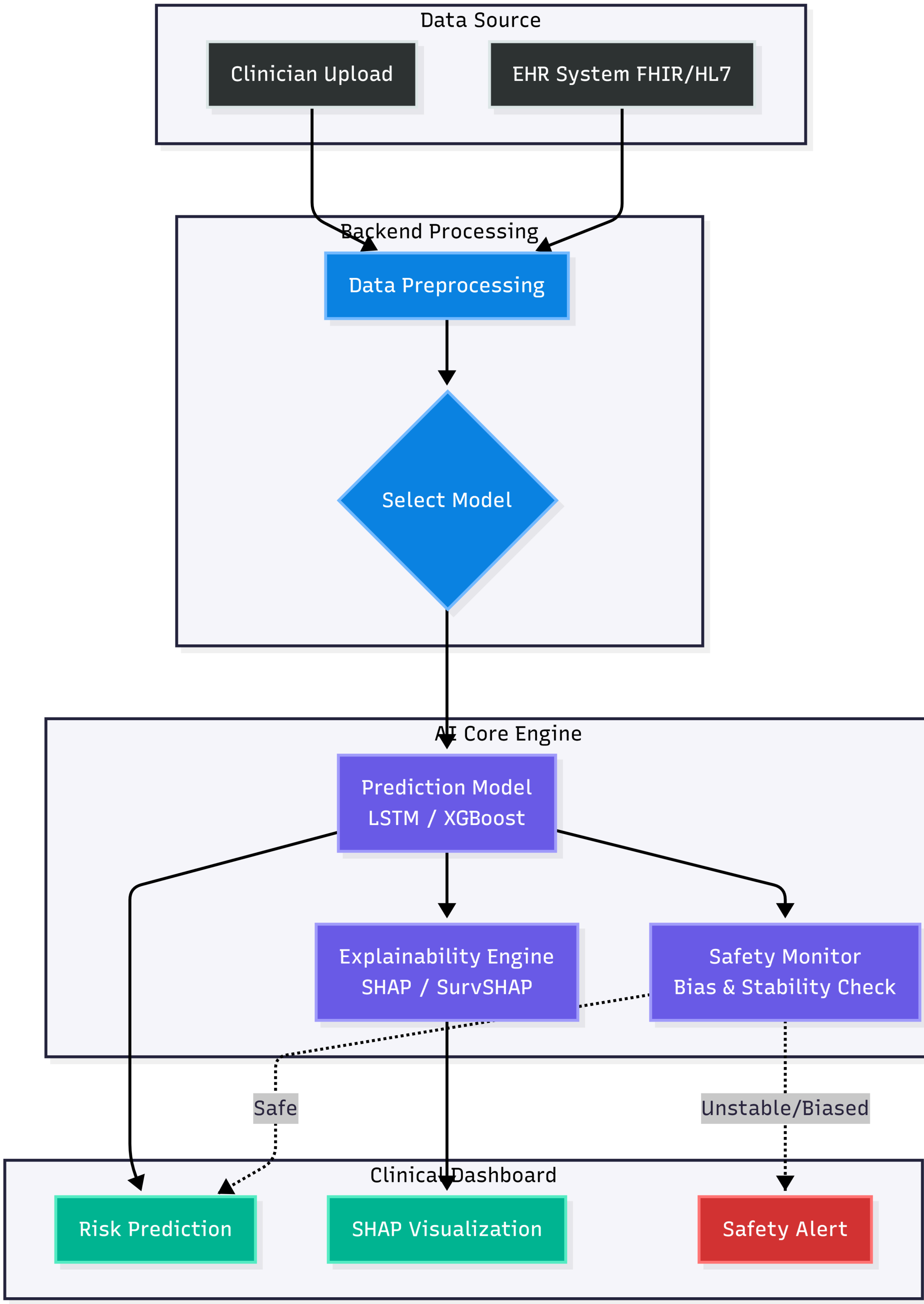
🫀

💊 SHAP-AID: Vision & Execution Board

1. Predictive Accuracy	2. Radical Transparency	3. Clinical Safety
Using LSTM & XGBoost to handle complex patient data (Time-series + Genomics).	No "Black Boxes." Every prediction must be backed by SHAP or SurvSHAP visuals.	Detect bias, flag unstable predictions, and warn clinicians of low-quality data.

Project Architecture



Data Source
Clinician Upload
EHR System FHIR/HL7
Backend Processing
Data Preprocessing
Select Model
AI Core Engine
Prediction Model
LSTM / XGBoost
Explainability Engine
SHAP / SurvSHAP
Safety Monitor
Bias & Stability Check
Safe
Unstable/Biased
Clinical Dashboard
Risk Prediction
SHAP Visualization
Safety Alert

Timeline

Phase 1: Discovery & Data Prep (Weeks 1–3)

Focus: Preparing the raw materials.

- [ ] **Literature Review:** Study `SurvSHAP(t)` and `CorrSHAP`.
- [ ] **Data Acquisition:** Download and explore datasets.
    - *Target:* NHANES (general health), TCGA (cancer), or OpenFDA.
- [ ] **Preprocessing Pipeline:** Handle missing values and normalize data.
- [ ] **Feature Selection:** Identify key features (age, BMI, CYP2D6 variants) and remove multicollinearity.

Phase 2: AI Engine Development (Weeks 4–7)

Focus: Building the backend logic.

- [ ] **Baseline Models:** Train simple models (Linear Regression, Random Forest) to set a benchmark.
- [ ] **Advanced Models:** Train XGBoost (tabular) and LSTM (time-series) on processed data.
- [ ] **SHAP Integration:** Implement `shap` library in Python to generate local explanations for predictions.
- [ ] **Model Comparison:** Compare accuracy vs. F1-score across all models.

Phase 3: Prototype & Dashboard (Weeks 8–11)

Focus: Building the UI/UX.

- [ ] **Frontend Setup:** Initialize React (or Streamlit for faster iteration).
- [ ] **Input Interface:** Build "Patient Upload" and "Model Selection" screens.
- [ ] **Visualization Components:** Create SHAP bar charts (red/blue bars) using a charting library (e.g., Recharts or Plotly).
- [ ] **Integration:** Connect the Python backend (Flask/FastAPI) to the frontend.

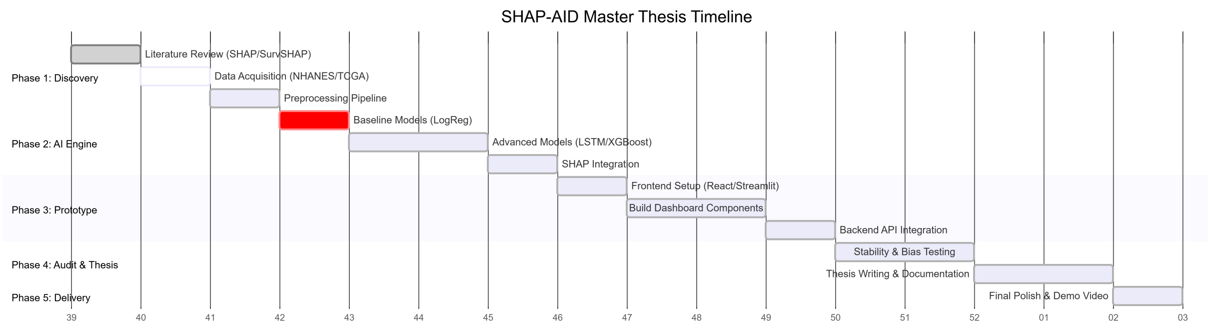
Phase 4: Audit & Thesis Evaluation (Weeks 12–14)

Focus: Meeting thesis requirements.

- [ ] **Stability Test:** Run 5-fold cross-validation and measure SHAP value changes (stability metric).
- [ ] **Bias Check:** Test the model across demographic groups to ensure fairness.
- [ ] **Trust Monitor:** Build logic for safety alerts (e.g., if prediction confidence is high but data is sparse, trigger alert).

Phase 5: Polish & Documentation (Weeks 15–16)

- [ ] **Final UI Polish:** Add dark mode styling and mobile responsiveness.
- [ ] **Thesis Writing:** Document architecture, stability results, and clinical relevance.
- [ ] **Demo Prep:** Record a video walkthrough of a patient scenario (e.g., oncology case).



Tech Stack & Resources

Core Technology

- **Language:** Python 3.9+
- **ML Libraries:** PyTorch/TensorFlow (for LSTM), XGBoost, Scikit-Learn
- **Explainability:** `shap`, `survshap` (Python packages)
- **Frontend:** React.js (custom UI) or Streamlit (rapid prototyping)
- **Backend:** FastAPI (recommended for ML) or Flask

Data Sources

- 📂 NHANES (CDC Data)
- 🧬 cBioPortal (Cancer Genomics)
- 🏥 MIMIC-III (ICU Data)

🧪 Experiments Tracker (Database)

Create a Notion Database with these columns to track your ML experiments.

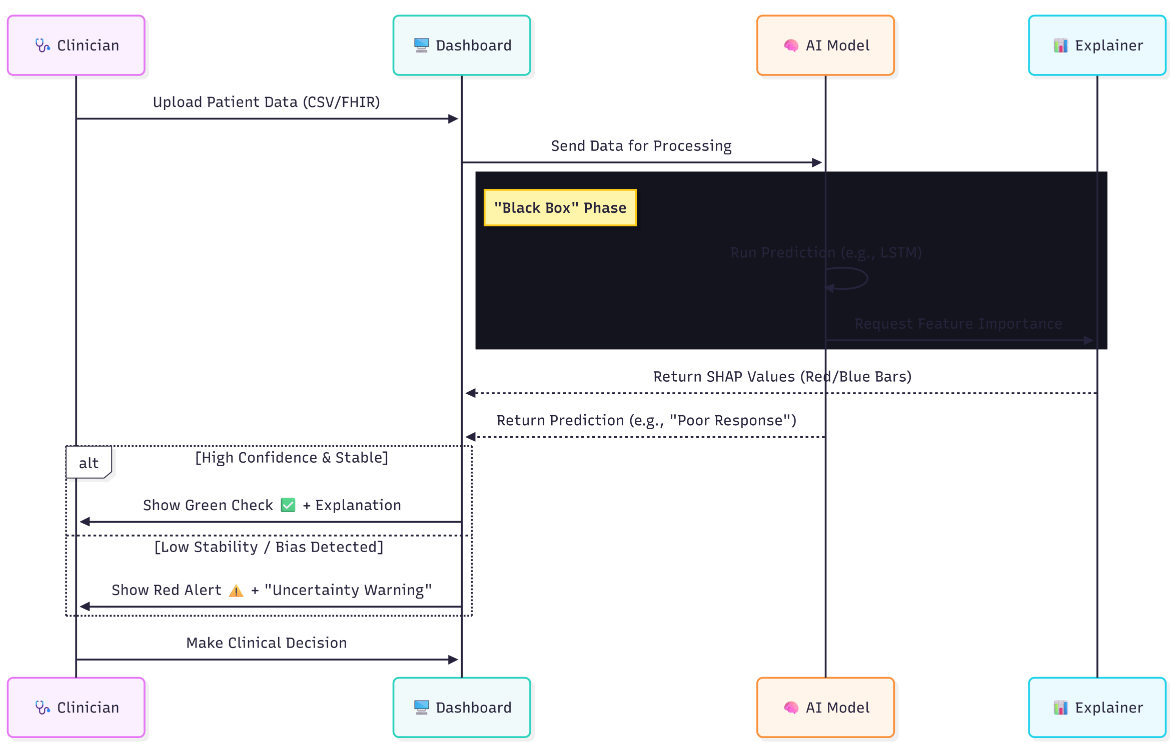
Experiment Name	Model Type	Accuracy	F1 Score	SHAP Stability	Notes
Baseline_LogReg	Logistic Regression	72%	0.68	High	Good baseline, but misses non-linear patterns.

XGBoost_v1	XGBoost	89%	0.86	Medium	Strong performance, but heavily relies on Age.
LSTM_TimeSeries	LSTM	94%	0.91	Low	Excellent prediction, but SHAP values fluctuate (Unstable).

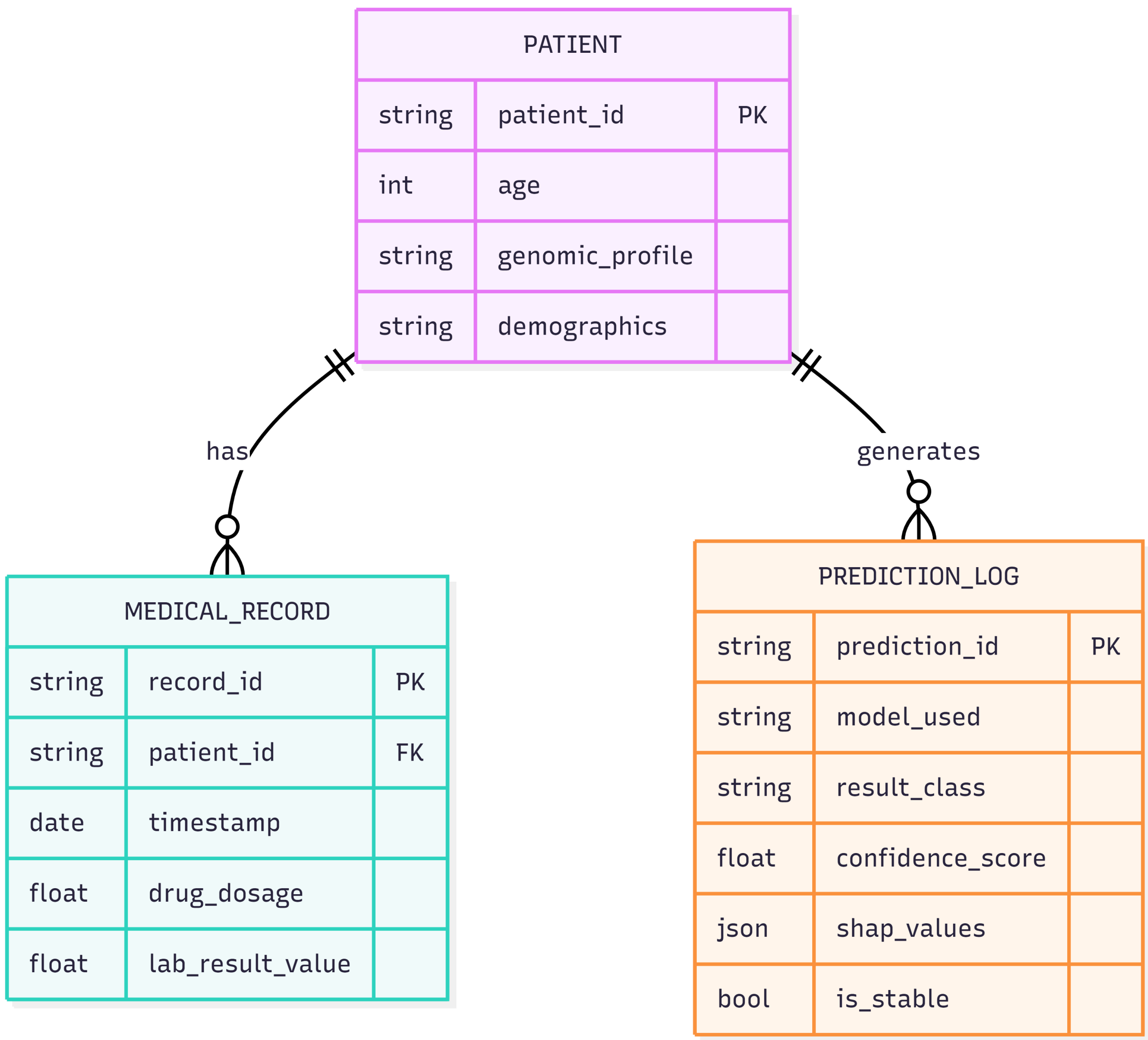
Startup & Scale Potential (Future Ideas)

- **Integration:** Add FHIR/HL7 API support for hospital EHRs.
- **Commercial Model:** "Freemium" for research, Subscription for Clinics.
- **Mobile App:** A lightweight version for doctors on rounds.

User Journey Flow



Proposed DB structure


PATIENT
string
patient_id
PK
int
age
string
genomic_profile
string
demographics
has
generates
MEDICAL_RECORD
string
record_id
PK
string
patient_id
FK
date
timestamp
float
drug_dosage
float
lab_result_value
PREDICTION_LOG
string
prediction_id
PK
string
model_used
string
result_class
float
confidence_score
json
shap_values
bool
is_stable