**1. X-ray Analyzer (TB, Pneumonia Detection)**

**Pre-trained Open Models**

* **TorchXRayVision** (NIH/Harvard) → PyTorch library with pre-trained CNNs (DenseNet, ResNet) on NIH ChestXray14, MIMIC-CXR.  
  👉 [TorchXRayVision GitHub](https://github.com/mlmed/torchxrayvision)
* **COVID-Net** (DarwinAI) → CNN for pneumonia/COVID detection, open-source.  
  👉 [COVID-Net](https://github.com/lindawangg/COVID-Net)
* **CheXNet (DenseNet-121)** → Pretrained on ChestXray14 dataset, detects 14 pathologies including pneumonia.

**Strategy**

✅ Use **TorchXRayVision** → it provides **already fine-tuned models for TB & pneumonia**.  
➡ No need to train from scratch → just fine-tune with TB subsets (Shenzhen/Montgomery datasets).

**🔹 2. Prescription OCR (Urdu + English)**

**Pre-trained/Open Models**

* **Tesseract OCR** (Google) → supports 100+ languages, including Urdu (urd.traineddata) and English (eng.traineddata).  
  👉 [Tesseract GitHub](https://github.com/tesseract-ocr/tesseract)
* **EasyOCR** (PyTorch-based) → pre-trained on 80+ languages including Urdu + English handwriting.  
  👉 [EasyOCR](https://github.com/JaidedAI/EasyOCR)
* **Kraken OCR** → works well with Arabic/Urdu script handwriting.

**Strategy**

✅ Start with **Tesseract** (fast, no training needed).  
➡ If results are poor → switch to **EasyOCR pretrained models** (already good for Urdu/English mixed text).

**🔹 3. Risk Scoring Model (Diabetes / Heart Disease)**

**Pre-trained/Open Models**

* No big “pre-trained” models directly for Pakistan — but **ready-to-use datasets** exist:
  + **PIMA Indians Diabetes Dataset** (UCI/Kaggle).
  + **UCI Heart Disease Dataset**.
* Pre-trained models are shared on Kaggle (Logistic Regression, Random Forest, XGBoost) → you can import weights or just replicate their pipelines.

**Strategy**

✅ Best path: take **Kaggle notebooks** (Logistic Regression, Random Forest) trained on PIMA/UCI → re-run quickly, then export model as .pkl.  
➡ You’ll have an MVP risk scorer in <1 day.

**🎯 Best Reuse Strategy for Techathon (14 Days)**

1. **X-ray Analyzer** → Use **TorchXRayVision (DenseNet-121 pretrained)** → fine-tune on TB/Pneumonia subsets.
2. **OCR** → Start with **Tesseract** (plug & play) → fallback to **EasyOCR pretrained weights**.
3. **Risk Scoring** → Import a **Kaggle-trained Logistic Regression/Random Forest model** on PIMA/UCI → export as .pkl → integrate via Flask.

✅ This way, you’ll avoid training from scratch and save ~1 week of work.  
✅ All are **open-source** and comply with Techathon guidelines.