**✅ Project Title**

**AI-Powered Injury Detection System Using Skeleton-Based Human Motion Analysis**

**🧠 Core Concept**

The system uses **computer vision** and **AI models** (based on deep learning and pose estimation) to detect and analyze the **skeleton movement** of athletes from video or image feeds, and determine if the motion patterns are consistent with known injury signatures (e.g., limp, imbalance, reduced range of motion).

**🧱 SYSTEM ARCHITECTURE**

**1. Frontend (Web Interface)**

* Upload or stream videos/images
* Real-time feedback on injury detection
* Player profile & injury history
* Admin dashboard (for coaches, physios, etc.)

**Tools:** React.js, Vue.js, or plain HTML/CSS + JavaScript

**2. Backend (Server-side)**

* Handle file upload and video streaming
* Trigger AI pipeline
* Manage user sessions and database

**Tools:** Python (Flask/FastAPI), Node.js (Express), Django

**3. AI/Machine Learning Module**

* Pose estimation (e.g., **OpenPose**, **MediaPipe**, **HRNet**, or **MoveNet**)
* Feature extraction from skeleton keypoints
* Classifier for injury detection (e.g., SVM, LSTM, CNN, or Transformer-based models)

**Steps:**

* Extract skeleton joints (17 or 33 points)
* Track and compare against normal vs abnormal patterns
* Use ML model trained on labeled injury/non-injury data

**4. Database**

* Store:
  + User data
  + Player videos & skeleton data
  + Model predictions
  + Injury history

**Tools:** PostgreSQL, MongoDB, or Firebase

**5. Cloud or Edge Deployment**

* Use GPU-powered cloud (e.g., Google Cloud, AWS, or Azure) for real-time inference
* OR use on-device TensorFlow.js or ONNX if local processing is needed

**🤖 AI/ML COMPONENTS DETAILS**

| **Task** | **Tool / Algorithm** |
| --- | --- |
| Skeleton Extraction | MediaPipe / OpenPose / MoveNet |
| Feature Processing | Normalized joint angles, gait cycle |
| Anomaly Detection | LSTM (temporal), SVM/CNN (static), Transformer (sequence) |
| Injury Classification | Trained classifier on labeled movement patterns |

**📈 Possible Injuries to Detect**

* Limping or gait imbalance (lower-body injuries)
* Limited range of motion in arms (shoulder injury)
* Slow or irregular movement (strain, muscle fatigue)
* Asymmetrical posture

**🔬 Datasets to Use (for training your model)**

* UPenn Gait Dataset
* [Injury Classification Dataset (custom or clinical)] – may need to annotate videos manually
* Use **transfer learning** from general human motion datasets (like Human3.6M, Kinetics, PoseTrack) and fine-tune for injury detection

**🔐 Additional Features (Optional)**

* Alert system for coaches
* Injury progression monitoring
* Player comparison
* Integration with wearable data (e.g., IMUs)

**🧪 Prototype MVP Scope**

Start small:

* Upload video
* Extract skeleton
* Analyze with ML model
* Display result: “Normal” or “Possible Injury”

Then grow from there.

**📦 Technologies Stack Summary**

| **Layer** | **Technologies** |
| --- | --- |
| Frontend | HTML/CSS/JS, React.js or Vue.js |
| Backend | Python (Flask or FastAPI) |
| AI Models | TensorFlow, PyTorch |
| Pose Estimation | MediaPipe, OpenPose, MoveNet |
| Database | PostgreSQL or MongoDB |
| Deployment | Docker + AWS/GCP |
|  |  |

µ 