

# *AI Deadlift Coach*

全方位即時硬舉姿勢分析與訓練系統

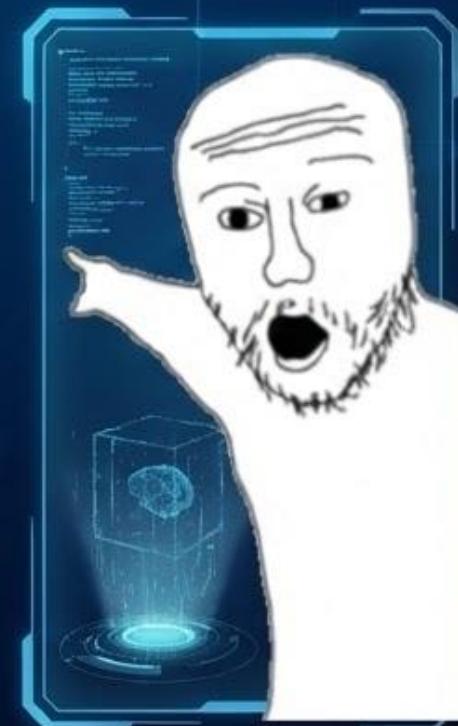
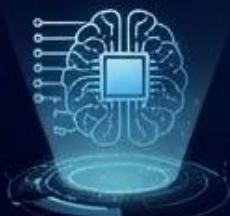
Cloud Computing Final Project



# 議程 (Agenda)

- 專案演進: MVP 與最終成品的差異
- 核心問題: 為什麼我們需要這個系統？
- 系統架構: 混合式 AI 引擎設計
- 核心技術: 醫學級演算法與機器學習

- 功能展示: 訓練追蹤與遊戲化
- Live Demo: 系統實際操作
- 雲端部署: Render 與 Docker 架構
- 未來展望: 下一步計畫



# 專案演進：從 MVP 到完整產品

## 產品迭代與功能升級

### 期中 MVP (Midterm)

Draft / Prototype

分析核心：基礎規則 (Rule-based)

分析模式：單一即時模式

互動回饋：簡單 Canvas 繪圖

功能深度：單次動作分析

部署環境：本地 Docker Compose



### 期末完整版 (Final)

Final Product / Advanced

分析核心：混合智能 (規則 + ML Random Forest)

分析模式：3 種模式 (即時 / AI / 混合)

互動回饋：語音教練、音效、多層級視覺警告

功能深度：訓練歷史、組數追蹤、遊戲化成就

部署環境：Render 雲端自動化部署 CI/CD



# 核心問題與動機



## 高風險傷害

硬舉動作若出現圓背或過早伸髋，極易造成不可逆的脊椎傷害。自我訓練難以察覺細微錯誤。



## 缺乏即時性

傳統錄影回放無法提供「當下」的修正建議。我們需要毫秒級的延遲回饋。



## 專案目標

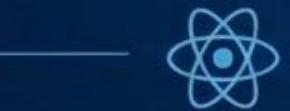
打造低成本、醫學級精準度的 AI 私人教練。

“Why should we hire you?”

me :



# 系統架構: 混合式 AI 引擎



## Frontend (React 19)

- 負責即時影像擷取、MediaPipe 骨架追蹤、規則運算 (Rule Engine) 與使用介面渲染。



## Backend (FastAPI)

- 處理複雜運算、ML 模型推論 (Random Forest)、以及 Session 狀態管理。



## Deployment

- Docker 容器化架構，部署於 Render PaaS，實現自動化 CI/CD 流程。



# 創新功能: 三種分析模式



## 1. 即時模式 (Real-time)

純前端計算，零 API 延遲。  
利用幾何規則引擎進行脊椎  
曲率與角度分析。適合網路  
不穩環境。



## 2. AI 模式 (ML)

純後端計算。將 30 幀動作序  
列送入 Random Forest 模  
型，識別 9 種複雜錯誤標籤  
(如：槓鈴離身太遠)。



## 3. 混合模式 (Hybrid Mode) - 推薦

結合兩者優勢：以即時規則  
引擎為優先回饋，後端 ML  
模型進行輔助確認，達到速  
度與精準度的完美平衡。

# 技術細節：醫學級演算法

## 1. 脊椎曲率計算 (Spine Curvature)

- 不同於傳統的肩臂連線，我們計算上下段脊椎向量的夾角：

```
curvature = arccos(dot(upper_spine, lower_spine))
```

- 有效區分「正確的臍紋罐前傾」與「危險的圓背」。  



## 2. Random Forest ML 模型

 特徵提取：30 帀滑動視窗 (Sliding Window)，包含角度、向量與統計特徵。

 實作方式：集成多棵決策樹學習硬舉動作中不同錯誤姿勢的非線性關係

 識別標籤：圓背、體提早上升、鎖膝過早等 9 種錯誤。



# 使用者體驗：訓練追蹤與遊戲化



## 語音教練

使用 Web Speech API 進行即時中文語音提示：「注意背部打直」、「膝蓋不要內夾」，減少使用者看螢幕的負擔。



## 量化數據

即時計算 Reps、組數 (Sets)、以及每次動作的 0-100 分評分。提供組間休息計時器。



## 成就系統

透過 LocalStorage 記錄生涯數據，解鎖「初學者」、「訓練狂人」等徽章，提升訓練黏著度。



# 資料流程 (Data Flow)



## 1. Capture

Webcam 擷取影像，每 33ms  
進行一次前端運算。



## 2. Process

前端計算脊椎角度；同時每 30  
幀打包數據送往後端。



## 3. Inference

後端 RF 模型分析序列特徵，  
回傳 9 種姿勢標籤。



## 4. Feedback

UI 融合兩者結果，觸發視覺  
警告與語音提示。



```
class DeadliftFeatureExtractor:
    def __init__(self):
        self.pose = mp.solutions.pose.Pose(
            static_image_mode=False,
            model_complexity=2,
            min_detection_confidence=0.3,
            min_tracking_confidence=0.3
        )

    # 計算兩點距離
    def dist(self, a, b):
        return np.sqrt((a[0]-b[0])**2 + (a[1]-b[1])**2)

    # 計算三點角度
    def calculate_angle(self, a, b, c):
        a, b, c = np.array(a), np.array(b), np.array(c)
        ba, bc = a - b, b - c
        cos_angle = np.dot(ba, bc) / ((np.linalg.norm(ba) * np.linalg.norm(bc)) + 1e-7)
        return np.degrees(np.arccos(np.clip(cos_angle, -1.0, 1.0)))

    def get_landmarks(self, results):
        if not results.pose_landmarks:
            return None
        lm = results.pose_landmarks.landmark
        return {
            'left_ear': [lm[7].x, lm[7].y],
            'left_shoulder': [lm[11].x, lm[11].y],
            'right_shoulder': [lm[12].x, lm[12].y],
            'left_elbow': [lm[13].x, lm[13].y],
            'right_elbow': [lm[14].x, lm[14].y],
            'left_wrist': [lm[15].x, lm[15].y],
            'right_wrist': [lm[16].x, lm[16].y],
            'left_knee': [lm[25].x, lm[25].y],
            'right_knee': [lm[26].x, lm[26].y],
            'left_ankle': [lm[27].x, lm[27].y],
            'right_ankle': [lm[28].x, lm[28].y]
        }
```



# 雲端部署架構



# Live Demo

展示即時分析、語音提示與成就解鎖功能



# 總結與未來展望

## 專案成果 🏆

我們成功從一個簡單的規則系統，進化為包含 ML 預測、語音互動與遊戲化元素的完整產品。

驗證了「混合式 AI」在運動科技的可行性。



## 未來展望



**多角度分析：**  
支援側面與正面同時分析。



**社群功能：**  
好友排行榜與訓練菜單分享。



**邊緣計算：**  
將 ML 模型轉為 TensorFlow.js，實現完全離線執行。