PraxisForma System Architecture Document

1. Executive Summary

This document outlines the comprehensive system architecture for PraxisForma, an AI-powered mobile coaching platform that provides biomechanical analysis for youth athletes. The architecture prioritizes privacy-first design, mobile-optimized performance, modular sport-specific analysis bots, and scalable international deployment.

2. Architecture Principles

2.1 Core Design Principles

- **Privacy-First**: Local processing with optional cloud sync, automatic PII removal
- **Mobile-Native**: Optimized for smartphone-based video capture and analysis
- **Modular**: Sport-specific bots with shared infrastructure and consistent interfaces
- **Youth-Safe**: COPPA/GDPR compliant with parental controls and age-appropriate interactions
- **Coach-Amplifying**: Enhances human coaching rather than replacing coach relationships
- **Globally Scalable**: Multi-region deployment supporting US and EU operations

2.2 Quality Attributes

- **Performance**: Sub-30 second video analysis, 99.5% uptime
- **Scalability**: Support 100,000+ concurrent users, elastic cloud infrastructure
- **Security**: End-to-end encryption, SOC 2 compliance, zero-trust architecture
- **Maintainability**: Clean separation of concerns, comprehensive testing, clear documentation
- **Usability**: Intuitive interfaces for athletes ages 12-18 and coaching professionals

3. High-Level System Overview

mermaid

```
graph TB
    subgraph "Mobile Applications"
        A1[iOS App]
        A2[Android App]
        A3[Coach Web Dashboard]
    end

    subgraph "API Gateway & Load Balancer"
        B1[Azure API Management]
        B2[Load Balancer]
    end

    subgraph "Core Services"
        C1[Authentication Service]
        C2[User Management Service]
        C3[Video Processing Service]
        C4[AI Analysis Service]
        C5[Coaching Service]
        C6[Progress Tracking Service]
        C7[Notification Service]
    end

    subgraph "AI/ML Infrastructure"
        D1[Azure Computer Vision]
        D2[Custom ML Models]
        D3[Model Training Pipeline]
        D4[Sport-Specific Scoring Engines]
    end

    subgraph "Data Layer"
        E1[PostgreSQL - User Data]
        E2[Azure Blob Storage - Videos]
        E3[Redis - Cache & Sessions]
        E4[InfluxDB - Analytics]
    end

    subgraph "External Integrations"
        F1[Payment Processing]
        F2[Email/SMS Services]
        F3[Analytics Platforms]
        F4[Customer Support]
    end
```

```
    A1 --> B1
    A2 --> B1
    A3 --> B1
    B1 --> B2
    B2 --> C1
    B2 --> C2
    B2 --> C3
    B2 --> C4
    B2 --> C5
    B2 --> C6
    B2 --> C7
    C3 --> D1
    C4 --> D2
    C4 --> D4
    D3 --> D2
    C1 --> E1
    C2 --> E1
    C5 --> E1
    C6 --> E1
    C3 --> E2
    C1 --> E3
    C6 --> E4
    C2 --> F1
    C7 --> F2
    C6 --> F3
    C2 --> F4
```

4. Component Architecture

4.1 Mobile Applications

Technology Stack:

- **iOS**: Swift/SwiftUI with React Native Bridge for shared components
- **Android**: Kotlin with React Native Bridge for shared components
- **Shared Logic**: React Native with TypeScript for business logic and UI components

Key Components:

- **Video Capture Module**: Camera integration with guided recording assistance
- **Local Analysis Engine**: On-device pose detection for immediate feedback
- **Sync Manager**: Intelligent cloud synchronization with conflict resolution

- **Coach Communication**: Real-time messaging and progress sharing
- **Privacy Manager**: Local face/body blurring and PII protection

Offline Capabilities:

- Complete video analysis functionality without internet connectivity
- Local storage of analysis results and progress data
- Intelligent sync queue for when connectivity returns
- Offline mode indicators and graceful degradation

4.2 API Gateway & Security Layer

Technology Stack:

- **API Gateway**: Azure API Management with custom policies
- **Load Balancing**: Azure Application Gateway with SSL termination
- **Authentication**: OAuth 2.0 with Azure Active Directory B2C
- **Authorization**: Role-based access control with custom claims

Security Features:

- JWT token validation with automatic refresh
- Rate limiting and DDoS protection
- Request/response transformation for privacy protection
- Comprehensive API logging and monitoring

4.3 Core Microservices

4.3.1 Authentication Service

Responsibilities:

- User registration and login workflows
- Multi-factor authentication for coaches and administrators
- Parental consent management for youth athletes
- Session management and token lifecycle

Technology: Node.js with TypeScript, Express framework **Database**: PostgreSQL for user credentials, Redis for sessions **External Dependencies**: Azure AD B2C, SendGrid for email verification

4.3.2 User Management Service

Responsibilities:

- User profile management (athletes, coaches, parents, administrators)
- Team and group organization
- Permission and role management
- Subscription and billing integration

Technology: Node.js with TypeScript, GraphQL API **Database**: PostgreSQL with row-level security **External Dependencies**: Stripe for subscription management

4.3.3 Video Processing Service

Responsibilities:

- Video upload handling and validation
- Automatic PII detection and blurring
- Video format standardization and optimization
- Secure storage and retrieval

Technology: Python with FastAPI, OpenCV for video processing **Storage**: Azure Blob Storage with encryption at rest **Processing**: Azure Container Instances for scalable video processing

4.3.4 AI Analysis Service

Responsibilities:

- Pose detection and biomechanical analysis
- Sport-specific scoring algorithm execution
- Coaching recommendation generation
- Analysis result caching and optimization

Technology: Python with FastAPI, PyTorch for ML inference **ML Infrastructure**: Azure Machine Learning for model deployment **Cache**: Redis for analysis result caching

4.3.5 Coaching Service

Responsibilities:

- Personalized coaching plan generation
- Drill recommendation engine
- Progress tracking and goal management

- Coach-athlete communication facilitation

Technology: Node.js with TypeScript, NestJS framework **Database**: PostgreSQL for coaching data **AI Integration**: OpenAI GPT for natural language coaching feedback

4.3.6 Progress Tracking Service

Responsibilities:

- Historical performance analysis
- Trend identification and visualization
- Goal setting and achievement tracking
- Comparative analytics

Technology: Python with FastAPI, Pandas for data analysis **Database**: InfluxDB for time-series data, PostgreSQL for metadata **Visualization**: Chart.js integration for mobile apps

4.3.7 Notification Service

Responsibilities:

- Push notification delivery
- Email and SMS communication
- In-app messaging
- Notification preference management

Technology: Node.js with TypeScript, Bull Queue for job processing **Infrastructure**: Azure Notification Hubs, SendGrid, Twilio **Database**: Redis for notification queues

4.4 AI/ML Infrastructure

4.4.1 Computer Vision Pipeline

```
mermaid

graph LR
    A[Video Upload] --> B[Frame Extraction]
    B --> C[Pose Detection]
    C --> D[Privacy Filtering]
    D --> E[Biomechanical Analysis]
    E --> F[Sport-Specific Scoring]
    F --> G[Coaching Recommendations]
    G --> H[Result Storage]
```

Technology Stack:

- **Pose Detection**: Azure Computer Vision API with custom pose models
- **Biomechanical Analysis**: Custom PyTorch models trained on sport-specific datasets
- **Privacy Protection**: OpenCV-based face detection and blurring
- **Model Serving**: Azure Machine Learning endpoints with auto-scaling

4.4.2 Sport-Specific Scoring Engines

PowerQuotient Score (PQS) - Shot Put/Discus:

- Release angle optimization analysis
- Power transfer efficiency calculation
- Footwork and rotation technique scoring
- Comparative performance benchmarking

LiftQuotient Score (LQS) - Strength Training:

- Movement pattern analysis for major lifts
- Safety assessment and injury risk calculation
- Progressive overload recommendations
- Form consistency tracking

Future Scoring Systems:

- SprintQuotient Score (SQS) for running mechanics
- JumpQuotient Score (JQS) for jumping technique
- SwingQuotient Score (SwQS) for bat/club sports

4.4.3 Model Training Pipeline

Training Infrastructure:

- Azure Machine Learning for distributed training
- MLflow for experiment tracking and model versioning
- Automated model validation and performance monitoring
- Continuous integration for model updates

Data Pipeline:

- Anonymized movement data collection
- Automated data labeling with coach validation
- Synthetic data generation for rare movement patterns
- Privacy-preserving federated learning capabilities

5. Data Architecture

5.1 Data Flow Architecture

```
mermaid

graph TD
    A[Mobile App] --> B[API Gateway]
    B --> C[Video Processing Service]
    C --> D[Azure Blob Storage]
    C --> E[AI Analysis Service]
    E --> F[PostgreSQL]
    E --> G[InfluxDB]
    F --> H[Coaching Service]
    G --> I[Progress Tracking Service]
    I --> J[Analytics Dashboard]
    H --> K[Notification Service]
```

5.2 Database Schema Design

PostgreSQL - Primary Database:

- **Users**: Athletes, coaches, parents, administrators
- **Teams/Groups**: Organizational structures and memberships
- **Subscriptions**: Billing and feature access management
- **Coaching Plans**: Personalized training programs and recommendations
- **Analysis Results**: Biomechanical analysis outcomes and scores

InfluxDB - Time-Series Data:

- **Performance Metrics**: Historical scoring data and improvement trends
- **Usage Analytics**: App engagement and feature utilization
- **System Metrics**: API performance and system health monitoring

Azure Blob Storage - File Storage:

- **Original Videos**: Encrypted storage with automatic lifecycle management
- **Processed Videos**: Privacy-filtered videos with pose overlays
- **Analysis Artifacts**: Detailed biomechanical analysis files
- **Model Assets**: Trained ML models and configuration files

5.3 Data Privacy & Compliance

Privacy-First Design:

- Automatic PII detection and removal in video processing
- Local-first processing with optional cloud sync
- Anonymous movement data aggregation for model improvement
- User-controlled data retention with automatic deletion policies
- Zero-knowledge architecture where PraxisForma cannot identify athletes from movement data

COPPA/GDPR Compliance:

- Explicit parental consent workflows for users under 13
- Right to be forgotten implementation with complete data purging
- Data portability features allowing users to export their information
- Privacy-by-design architecture with minimal data collection
- Regular compliance audits and documentation

6. Security Architecture

6.1 Security Layers

Application Security:

- OAuth 2.0 authentication with Azure AD B2C
- JWT tokens with automatic rotation and revocation
- Role-based access control with principle of least privilege
- Input validation and sanitization at all API endpoints
- SQL injection prevention through parameterized queries

Network Security:

- TLS 1.3 encryption for all API communications
- Web Application Firewall (WAF) with custom rules

- DDoS protection and rate limiting
- Network segmentation with private subnets
- VPN access for administrative functions

Data Security:

- Encryption at rest using Azure Key Vault managed keys
- Field-level encryption for sensitive personal data
- Secure key rotation and management
- Database encryption with Transparent Data Encryption (TDE)
- Backup encryption and secure storage

Infrastructure Security:

- Zero-trust network architecture
- Multi-factor authentication for all administrative access
- Regular security scanning and vulnerability assessments
- Immutable infrastructure with infrastructure-as-code
- Comprehensive audit logging and monitoring

6.2 Threat Model & Mitigation

Identified Threats:

- Unauthorized access to youth athlete data
- Video interception during upload/analysis
- Model poisoning attacks on AI systems
- Account takeover and identity theft
- Data exfiltration and privacy breaches

Mitigation Strategies:

- End-to-end encryption for video uploads
- Anomaly detection for unusual access patterns
- Model validation and adversarial training
- Multi-factor authentication and session monitoring
- Data loss prevention (DLP) tools and policies

7. Performance & Scalability

7.1 Performance Requirements

Response Time Targets:

- Video upload initiation: < 2 seconds
- Pose detection completion: < 15 seconds
- Scoring analysis completion: < 30 seconds
- API response times: < 200ms for 95th percentile
- Mobile app launch time: < 3 seconds

Throughput Requirements:

- Support 1,000+ concurrent video uploads
- Process 10,000+ analyses per hour during peak times
- Handle 100,000+ daily active users
- Manage 1M+ API requests per hour

7.2 Scalability Strategy

Horizontal Scaling:

- Microservices architecture with independent scaling
- Container orchestration using Azure Kubernetes Service
- Auto-scaling based on CPU, memory, and queue depth metrics
- Load balancing across multiple regions

Vertical Scaling:

- GPU-accelerated instances for AI/ML workloads
- Memory-optimized instances for video processing
- Compute-optimized instances for API services
- Storage-optimized instances for database workloads

Caching Strategy:

- Redis for session and frequently accessed data
- CDN for static assets and processed videos
- Application-level caching for expensive computations

- Database query optimization and indexing

8. International Deployment Architecture

8.1 Multi-Region Strategy

Primary Regions:

- **US East**: Primary deployment for North American users
- **EU West**: European deployment for GDPR compliance and performance
- **Future Regions**: Asia-Pacific based on market expansion

Data Residency:

- User data stored in region of user registration
- Cross-region replication for disaster recovery
- Local processing requirements for privacy compliance
- Region-specific feature configurations

8.2 Portugal/EU Specific Considerations

GDPR Compliance:

- EU-specific data processing agreements
- Local data residency in EU data centers
- Enhanced privacy controls and user rights
- Explicit consent mechanisms for data processing

Localization Requirements:

- Multi-language support (Portuguese, Spanish, French, German)
- Local payment methods and currencies
- Cultural adaptation of coaching methodologies
- Integration with European sports federations

9. Monitoring & Observability

9.1 Application Monitoring

Key Metrics:

- Video processing success rates and latency

- AI model accuracy and performance metrics
- User engagement and retention analytics
- API performance and error rates
- Mobile app crash rates and performance

Monitoring Tools:

- Azure Application Insights for application telemetry
- Azure Monitor for infrastructure metrics
- Custom dashboards for business metrics
- Real-time alerting for critical issues
- Performance profiling and optimization

9.2 Business Intelligence

Analytics Pipeline:

- Real-time event streaming with Azure Event Hubs
- Data warehouse using Azure Synapse Analytics
- Business intelligence dashboards with Power BI
- Machine learning insights for product optimization
- Predictive analytics for user behavior

10. Disaster Recovery & Business Continuity

10.1 Backup Strategy

Data Backup:

- Automated daily backups of all databases
- Cross-region backup replication
- Point-in-time recovery capabilities
- Encrypted backup storage with lifecycle management

Application Backup:

- Infrastructure-as-code for rapid environment recreation
- Container image registry with version control
- Configuration management and secret backup

- Automated deployment pipeline for rapid restoration

10.2 Disaster Recovery Plan

Recovery Time Objectives (RTO):

- Critical services: 1 hour
- Non-critical services: 4 hours
- Full service restoration: 24 hours

Recovery Point Objectives (RPO):

- User data: 15 minutes
- Video content: 1 hour
- System configurations: 24 hours

Failover Procedures:

- Automated health checks and failover triggers
- Manual failover procedures for edge cases
- Data synchronization and consistency verification
- User communication and status page updates

11. Development & Deployment

11.1 Development Environment

Local Development:

- Docker Compose for local service orchestration
- Mock services for external dependencies
- Automated testing environment setup
- Hot-reload capabilities for rapid development

Staging Environment:

- Production-like environment for integration testing
- Automated deployment from feature branches
- Performance testing and load simulation
- Security scanning and vulnerability assessment

11.2 CI/CD Pipeline

Continuous Integration:

- Automated testing on every commit
- Code quality checks and linting
- Security scanning and dependency auditing
- Performance regression testing

Continuous Deployment:

- Blue-green deployment strategy
- Automated rollback on failure detection
- Feature flags for gradual feature rollout
- Canary deployments for high-risk changes

12. Cost Optimization

12.1 Resource Optimization

Compute Optimization:

- Auto-scaling to match demand patterns
- Reserved instances for predictable workloads
- Spot instances for batch processing jobs
- Right-sizing based on actual usage metrics

Storage Optimization:

- Automated data lifecycle management
- Compression and deduplication for video storage
- Cold storage tiers for archival data
- Intelligent tiering based on access patterns

12.2 Cost Monitoring

Cost Tracking:

- Real-time cost monitoring and alerting
- Cost allocation by service and feature

- Budget controls and spending limits
- Regular cost optimization reviews

13. Technology Stack Summary

13.1 Frontend Technologies

- **Mobile**: React Native with TypeScript
- **Web Dashboard**: React with TypeScript
- **State Management**: Redux Toolkit
- **UI Components**: Native Base, Chakra UI
- **Testing**: Jest, React Testing Library, Detox

13.2 Backend Technologies

- **API Services**: Node.js with TypeScript, Express/NestJS
- **AI/ML Services**: Python with FastAPI, PyTorch
- **Message Queues**: Redis with Bull Queue
- **Authentication**: Azure AD B2C with OAuth 2.0
- **Testing**: Jest, pytest, Supertest

13.3 Infrastructure Technologies

- **Cloud Platform**: Microsoft Azure
- **Container Orchestration**: Azure Kubernetes Service
- **Databases**: PostgreSQL, Redis, InfluxDB
- **Storage**: Azure Blob Storage with encryption
- **Monitoring**: Azure Monitor, Application Insights
- **CI/CD**: Azure DevOps, GitHub Actions

13.4 AI/ML Technologies

- **Computer Vision**: Azure Computer Vision API
- **Machine Learning**: Azure Machine Learning
- **Model Framework**: PyTorch, ONNX Runtime
- **Experiment Tracking**: MLflow
- **Model Serving**: Azure ML Endpoints

14. Conclusion

This architecture provides a robust, scalable, and secure foundation for PraxisForma's AI-powered coaching platform. The design emphasizes privacy protection for youth athletes, mobile-optimized performance, and international scalability while maintaining the flexibility to add new sports and coaching methodologies.

Key architectural strengths include:

- Privacy-first design with local processing capabilities
- Modular architecture enabling rapid sport expansion
- Comprehensive security and compliance measures
- Scalable infrastructure supporting global deployment
- Robust monitoring and observability
- Cost-optimized resource utilization

The architecture supports PraxisForma's mission to democratize elite athletic coaching while ensuring the safety, privacy, and positive development of youth athletes worldwide.

Document Version: 1.0
Last Updated: [Current Date]
Document Owner: PraxisForma Engineering Team
Review Cycle: Quarterly architectural reviews with monthly updates