Deployment Guide: Mobile Disease Detection App

Production Readiness Assessment

Current Prototype Capabilities

The app utilizes computer vision to detect diseases from images.

Basic user interface for capturing and analyzing images.

Limitations that Need Addressing

Limited dataset for training may affect accuracy.

User interface may not be intuitive for all users.

Lack of offline functionality.

Required Improvements for Real-World Use

Expand the dataset for diverse disease detection.

Enhance user interface based on user feedback.

Implement offline capabilities for areas with poor connectivity.

Infrastructure Requirements

Minimum System Requirements for Production

Operating System: Android/iOS

RAM: 2 GB

Storage: 100 MB free space

Recommended Hardware Specifications

Operating System: Latest Android/iOS version

RAM: 4 GB or more

Storage: 500 MB free space

Camera: Minimum 8 MP

Network and Connectivity Needs

Internet access for initial model download and updates.

Ability to function offline for data collection.

Security Infrastructure Requirements

Secure data transmission protocols (e.g., HTTPS).

Regular security audits and vulnerability assessments.

Integration Planning

How This AI Fits into Existing Humanitarian Systems

Integrates with health information systems for data sharing.

Provides real-time disease detection to support health workers.

Data Flow and Workflow Integration

Capture image → Process through AI model → Display results → Log data to health system.

User Training Requirements

Training sessions for health workers on app usage.

User manuals and quick reference guides.

Change Management Considerations

Engage stakeholders early to gather feedback.

Communicate benefits and changes to all users.

Security and Privacy

Data Protection Measures Needed

Encrypt sensitive data both in transit and at rest.

Anonymize user data to protect identities.

User Access Controls

Role-based access for different user levels (e.g., admin, health worker).

Regularly update access permissions.

Audit and Compliance Requirements

Ensure compliance with local data protection regulations.

Maintain logs of data access and modifications.

Risk Mitigation Strategies

Regularly update software to patch vulnerabilities.

Conduct user training on security best practices.

Scaling Considerations

Expected User Load and Data Volume

Estimate user base growth based on deployment area.

Plan for increased data storage needs.

Performance Optimization Needs

Optimize model inference time for faster results.

Monitor app performance and user feedback for improvements.

Multi-Location Deployment Planning

Prepare for regional variations in disease prevalence.

Localize app content and language as needed.

Backup and Disaster Recovery

Implement regular data backups.

Develop a disaster recovery plan for data loss scenarios.

Maintenance and Support

Ongoing Monitoring Requirements

Monitor app performance and user engagement metrics.

Regularly review model accuracy and update as needed.

Update and Maintenance Schedules

Schedule regular updates for app features and security.

Plan for periodic model retraining with new data.

User Support Infrastructure

Establish a helpdesk for user inquiries.

Provide online resources and FAQs.

Performance Tracking and Optimization

Use analytics to track user interactions and app performance.

Gather user feedback for continuous improvement.

Cost Planning

Initial Deployment Costs

Development and testing costs.

Hardware and software procurement.

Ongoing Operational Expenses

Server hosting and maintenance.

User support and training costs.

Training and Support Costs

Budget for training sessions and materials.

Allocate funds for ongoing user support.

Return on Investment Considerations

Evaluate cost savings from improved disease detection.

Measure impact on health outcomes and resource allocation.

Implementation Timeline

Recommended Deployment Phases

Phase 1: Prototype testing and feedback collection.

Phase 2: Initial deployment in select areas.

Phase 3: Full-scale rollout and monitoring.

Key Milestones and Deliverables

Completion of user training.

Launch of the app in target regions.

First round of user feedback and adjustments.

Risk Factors and Contingencies

Identify potential technical issues and user resistance.

Develop contingency plans for deployment delays.

Success Metrics and Evaluation

Track user adoption rates and satisfaction.

Measure accuracy of disease detection and impact on health outcomes.

Working with Technical Teams

Information to Provide to Developers

Detailed user requirements and use cases.

Access to existing health data systems for integration.

Key Decisions for Technical Implementation

Choose appropriate technology stack for app development.

Determine model training and deployment strategies.

Humanitarian Requirements Specification

Ensure alignment with humanitarian goals and user needs.

Address cultural and contextual factors in app design.

Quality Assurance and Testing Protocols

Establish testing protocols for functionality and usability.

Conduct field tests with end-users to validate performance.