Technical Handover Package: Mobile Disease Detection App

Executive Summary

Project Overview and Humanitarian Objectives: The AI for Early Detection of Crop Diseases project aims to empower farmers by providing a mobile application that utilizes computer vision to detect early signs of crop diseases from smartphone photos. This initiative seeks to reduce crop losses and enhance food security in vulnerable communities.

Technical Approach and Rationale: The solution leverages a lightweight computer vision model optimized for mobile devices, enabling real-time analysis of images taken by farmers. This approach ensures accessibility and immediate feedback, crucial for timely interventions.

Key Implementation Decisions: The decision to implement a hybrid processing model allows for initial analysis on smartphones, with periodic updates and improvements synced to organizational servers. This balances performance with resource constraints.

Production Requirements

Security Requirements

Specific Security Measures for Farmers Data:

- Implement end-to-end encryption for data transmission.
- Use anonymization techniques for any stored data to protect farmer identities.

Infrastructure Security for Mobile Optimized:

- Utilize secure cloud services with robust access controls.
- Regularly update and patch all software components to mitigate vulnerabilities.

Compliance Requirements for Humanitarian Contexts:

- Adhere to GDPR and local data protection regulations.
- Establish clear data usage policies communicated to farmers.

Performance and Scalability

Expected Performance Benchmarks for Computer Vision:

Aim for at least 90% accuracy in disease detection on a validation dataset.

Ensure response time for image analysis is under 5 seconds on mobile devices.

Scalability Requirements for Mobile Optimized:

Design the architecture to support up to 10,000 concurrent users.

Implement load balancing to manage traffic spikes effectively.

Resource Allocation Guidelines:

Allocate 2 CPU cores and 4GB RAM per instance for the mobile backend.

Ensure sufficient storage (minimum 100GB) for image data on organizational servers.

Error Handling and Resilience

Specific Error Patterns for Computer Vision Failures:

Identify common misclassifications and implement fallback mechanisms.

Log errors for analysis and model retraining.

Graceful Degradation Strategies:

Provide users with generic health status messages if the model fails.

Allow users to manually report suspected diseases as a fallback.

User-Friendly Error Messaging for Humanitarian Contexts:

Use simple language and visuals in error messages to ensure understanding.

Offer guidance on next steps if the app fails to analyze an image.

Development Team Requirements

Skills and Expertise

Required Technical Skills for Computer Vision Implementation:

Proficiency in TensorFlow or PyTorch for model development.

Experience with mobile app development (iOS/Android).

Humanitarian Domain Knowledge Needs:

- Understanding of agricultural practices and crop disease management.
- Familiarity with the challenges faced by farmers in target regions.

Team Composition Recommendations:

- 1 Project Manager
- 2 Data Scientists (Computer Vision specialists)
- 2 Mobile Developers (1 iOS, 1 Android)
- 1 UX/UI Designer
- 1 Agricultural Expert

Infrastructure and Tools

Development Environment Requirements:

- Use Docker for containerization of the application.
- Set up a CI/CD pipeline using GitHub Actions or Jenkins.

Deployment Infrastructure for Mobile Optimized:

- Utilize AWS or Google Cloud for hosting the backend services.
- Ensure mobile app stores (Google Play, Apple App Store) are prepared for deployment.

Monitoring and Logging Tools:

- Implement Prometheus for performance monitoring.
- Use ELK Stack (Elasticsearch, Logstash, Kibana) for logging and analysis.

Implementation Timeline

Phase 1: Foundation (2 months)

- Core computer vision implementation
- Basic infrastructure setup
- Security framework implementation

Phase 2: Integration (2 months)

Phase 2: Integration (3 months)

- Humanitarian workflow integration
- User interface development
- Data pipeline implementation

Phase 3: Production (2 months)

- Production deployment for mobile optimized
- Performance optimization
- Monitoring implementation

Risk Assessment and Mitigation

Technical Risks

Computer Vision Specific Challenges:

Risk of model overfitting; mitigate by using diverse training datasets.

Mobile Optimized Deployment Risks:

Potential performance issues on low-end devices; optimize model size and complexity.

Mitigation Strategies:

Regularly update the model based on user feedback and new data.

Humanitarian Context Risks

Data Privacy and Protection Risks:

Ensure compliance with data protection laws; conduct regular audits.

Beneficiary Impact Risks:

Engage with farmers to understand their needs and adapt the tool accordingly.

Operational Continuity Risks:

Develop a disaster recovery plan to ensure service availability.

Monitoring and Maintenance

Performance Monitoring

Key Metrics for Computer Vision in Humanitarian Context:

Track accuracy, response time, and user engagement metrics.

Automated Monitoring Setup:

Use Grafana dashboards for real-time performance visualization.

Alert Thresholds and Responses:

Set alerts for performance degradation beyond 10% of expected accuracy.

Model Maintenance

Retraining Schedule and Procedures:

Schedule model retraining every 6 months or when significant new data is available.

Data Quality Monitoring:

Implement checks for image quality and relevance before processing.

Performance Degradation Detection:

Monitor model performance metrics continuously and trigger retraining if accuracy drops below 85%.

User Support and Training

User Training Requirements for Humanitarian Staff:

Conduct workshops on using the app and interpreting results.

Support Documentation Needs:

Create a comprehensive user manual and FAQs.

Feedback Collection Mechanisms:

Implement in-app feedback forms for continuous improvement.

Success Metrics and KPIs

Success Metrics and KPIs

Technical Metrics

Performance Benchmarks for Computer Vision:

Maintain 90% accuracy and 5-second response time.

System Availability and Reliability:

Target 99.9% uptime for the application.

Response Time Requirements:

Ensure image analysis is completed within 5 seconds.

Humanitarian Impact Metrics

Specific Impact Measures for Farmers:

Track reduction in crop losses reported by users.

Operational Efficiency Improvements:

Measure time saved in disease detection and response.

User Adoption and Satisfaction:

Aim for at least 70% user adoption within the first year.

Compliance and Documentation

Required Technical Documentation:

Maintain up-to-date API documentation and architecture diagrams.

Audit and Compliance Requirements:

Conduct bi-annual audits to ensure compliance with data protection regulations.

Change Management Procedures:

Establish a formal process for managing changes to the application and infrastructure.

Emergency Procedures

Emergency Procedures

System Failure Response Protocols:

Define escalation paths for technical issues and user-reported problems.

Data Breach Response Procedures:

Develop a response plan that includes notification protocols for affected users.

Rollback and Recovery Procedures:

Implement version control for the application to facilitate quick rollbacks in case of failures.