Request for Proposals – SERVICES

CONSULTANCY FOR INDEPENDENT QUALITY ASSURANCE (QA) AND TESTING OF MOBILE AND WEB APPLICATION

Issued by
The Global Alliance for Improved Nutrition (GAIN)

Table of Contents

1. Background

An Android-based mobile application and its companion web platform have been developed to support field data collection, storage, cleaning, and reporting. The application enables enumerators to capture complex, hierarchical data structures, such as:

i. Shop Reports: retail outlet details (location, GPS, outlet type, name and other related details).
ii. Food Reports: reports per food vehicle (e.g., Oil, Rice, Salt), created within each Shop Report.
iii. Brand Products: brand-level records within a Food Report.
iv. Packaging: packaging details (type, size, supplier, fortification labelling, price/stock).
v. Samples: sample-specific data (method, amount, production date, best-before date, batch number, and photos).

While the system is functional, significant technical and quality issues persist:

a. Food Reports sometimes fail to submit or only submit partially.
b. ID mismatches across sync result in duplication, orphaned records, or mislinked data.
c. Images and attachments often fail to upload or are mismatched.
d. Data corruption occurs under concurrent user submissions.
e. Navigation issues (e.g., back button) can delete unsaved inputs.

These problems undermine the reliability, scalability, and usability of the application. To address this, an independent quality assurance review is required. The objective is to ensure the codebase and overall system have been developed in a structured, lean, and professional manner, enabling maintainability, scalability, and smooth handover to future developers. The review should also verify that all features function as intended, the system is adequately documented, and that there are no technical issues that may compromise performance, reliability, or data integrity. Any additional risks should be identified and reported.

2. Objectives

The QA review aims to:

i. Assess the quality and structure of the source code for both mobile and web applications.
ii. Verify that the code is efficient, maintainable, and written to professional standards.
iii. Ensure that the application is easily transferable for future developers to update, extend, or maintain.
iv. Validate that all existing features including complex nested data entry, image capture, offline-first storage, and synchronization work according to specifications.
v. Check that documentation is sufficient to support continuity, verification, and onboarding.
vi. Identify any risks, weaknesses, or technical debt in the current implementation.
vii. Confirm that there are no interference issues or vulnerabilities that may compromise functionality, data storage, or system stability.

3. Scope of Work

3.1 Code Review

i. Evaluate code structure, readability, and maintainability.
ii. Check for lean/efficient coding practices (no unnecessary repetition, optimized logic).
iii. Verify adherence to industry standards and best practices (naming conventions, modular design, use of frameworks/libraries).
iv. Assess whether the codebase includes sufficient comments and narrative code for continuity.
v. Identify potential risks related to scalability, performance, or future maintenance or potential handover to other parties.

3.2 Functionality Testing

i. End-to-End Feature Validation: Verify that all application features operate as intended, including the complete hierarchical reporting workflow (Shop → Food Report → Brand Products → Packaging → Samples).
ii. Submission Workflows with Nested Data: Test report submission workflows under conditions involving large data volumes and deeply nested entities to ensure stability and accuracy.
iii. Offline-First Behaviour and Recovery: Validate offline-first data entry, confirming that records are preserved and fully recoverable after app crashes, back navigation, forced closures, or power loss.
iv. Synchronization Testing:
a. Web → Mobile: Ensure that all approved food brand products, along with their respective packaging details, sample, and associated images, are consistently synchronized from the web application to the mobile application without duplication, mismatch, or data loss.
b. Mobile → Web: Validate that each complete Food Report submitted from the mobile application, containing all brand products, their respective packagings, samples (with accurate sample counts), and linked images are transmitted accurately to the web application upon syncing, with no partial uploads of reports, ID mismatches of any details, or orphaned/disconnected records.
v. Simulate concurrent submissions by multiple users to detect data loss, duplication, or mismatches.
vi. Ensure data storage and processing are secure, reliable, and consistent.

3.3 Documentation Review

i. Technical Documentation Audit: Assess the completeness and accuracy of technical documentation, including installation guides, architecture overviews, dependency lists, configuration instructions, and deployment procedures.
ii. API Documentation: Verify that API documentation is available, up to date, and aligned with the current implementation, including request/response formats, error codes, authentication requirements, and versioning.
iii. User Guides and Operational Manuals: Check for the availability and accuracy of user-

facing documentation (e.g., enumerator user guides, operational manuals, troubleshooting guides), ensuring they support smooth onboarding and field usage.
iv. Testing Documentation: Confirm the presence of QA-related materials such as test plans, test cases, test scripts, bug reports, and defect logs.
v. Documentation Preparation and Improvement: Prepare or update missing documentation where gaps exist, and ensure all materials are consolidated into a clear, structured, and standardized format suitable for both technical teams and end users.

3.4 Risk Assessment

Identify weaknesses or gaps that may impact:

i. Security: vulnerabilities, lack of encryption, authentication flaws.
ii. Performance: inefficient queries, poor handling of large datasets, excessive resource use.
iii. Reliability: sync failures, partial submissions, crash risks, unstable integrations.
iv. Maintainability: poor documentation, lack of modularity, outdated dependencies.

Provide a clear overview of areas needing urgent attention.

4. Deliverables

The consultant/team will provide the following outputs at the end of the engagement:

4.1 QA Review Report

A comprehensive written report structured as follows:

i. Overview of Findings: High-level summary of the QA process, scope, test coverage, and outcomes.
ii. Strengths and Weaknesses: Detailed assessment of both mobile and web components, highlighting well-implemented aspects as well as areas requiring urgent remediation, including expected level of effort to adjust/fix.
iii. Areas Needing Remediation or Enhancement: Prioritized list of technical and functional fixes, categorized as critical, high, medium, and low severity.
iv. Identified Risks and Mitigation Strategies: Clear articulation of risks affecting system reliability, data integrity, security, and user experience, with corresponding technical recommendations (e.g., redesigning sync logic, introducing transactional submissions, enforcing referential integrity).

4.2 Feature Functionality Verification Log

i. Exhaustive log of all test cases executed, including:
a. Normal workflow scenarios (Shop → Food Reports → Brand Products → Packaging → Samples).
b. Edge cases (partial submissions, back navigation, app crashes, offline capture).
c. Sync scenarios (Web → Mobile, Mobile → Web, conflict resolution).
d. Concurrency scenarios (multiple simultaneous submissions).

ii. Each test case documented with test description, environment, expected result, actual result, pass/fail status, and evidence (screenshots, logs).

4.3 Code Quality Assessment

i. Findings from static code analysis and manual review.
ii. Evaluation of coding practices, adherence to standards, modularity, readability, and error-handling logic.
iii. Specific technical recommendations (e.g., refactoring strategies, dependency updates, API error handling improvements).

4.4 Documentation Gap Analysis

i. Audit of existing technical documentation, API references, deployment notes, and operational guides.
ii. Identification of missing documentation (e.g., test plans, sync protocol specifications).
iii. Recommendations for producing or updating documentation to industry-standard formats.

4.5 Final Presentation & Walkthrough

i. Documentation Preparation: Develop or update missing materials identified during the gap analysis, including but not limited to:
a. Technical documentation (installation guides, architecture overviews, dependency lists, configuration and deployment instructions).
b. API documentation aligned with the current implementation.
c. User guides and operational manuals to support onboarding and field use.
d. QA documentation (test plans, test cases, defect logs, bug reports).
ii. Final Presentation to Stakeholders: Deliver a structured presentation summarizing key findings, risks, recommendations, and remediation priorities.
iii. Test Case Walkthrough: Document or list the critical test cases where issues were identified, showing both expected and actual outcomes.
iv. Handover Package: Provide all deliverables, including the QA Review Report, Feature Functionality Verification Log, Code Quality Assessment, updated documentation, test scripts, logs, and all supporting materials generated during the engagement.

5. Methodology

The QA review will employ a combination of manual, automated, and analytical techniques to ensure comprehensive coverage:

5.1 Static Code Analysis

i. Automated scanning of mobile (Android) and web codebases using industry-standard tools (e.g., SonarQube, Lint, PMD).
ii. Manual review of key components (sync logic, database handling, API integration).
iii. Identification of unused code, redundant logic, outdated libraries, and insecure practices.

5.2 Functional Testing

i. Manual testing of end-to-end workflows across the hierarchical reporting model.
ii. Exploratory testing to uncover hidden issues in navigation, data capture, and error handling.
iii. Automated testing (where feasible) for regression and repetitive scenarios.
iv. Testing across low-end, mid-range, and high-end Android devices, with multiple OS versions.

5.3 Load & Concurrency Testing

i. Use of tools such as JMeter/Locust to simulate multiple concurrent users (50–200).
ii. Measurement of throughput, error rates, and server response under load.
iii. Validation of submission integrity (no duplication, missing data, or mismatches under stress).

5.4 Sync Validation

i. Testing Web → Mobile sync: ensure latest reference data propagates correctly.
ii. Testing Mobile → Web sync: confirm atomic submissions (no partial report uploads, images linked correctly).
iii. Validation of idempotency: repeated submissions do not create duplicates.

5.5 Documentation Audit

i. Review of technical documentation (architecture diagrams, deployment guides).
ii. Verification of API references against actual implementation.
iii. Assessment of developer onboarding materials.

5.6 Developer Interviews/Discussions

i. Where possible, discussions with current developers to validate assumptions about design choices, known issues, and legacy constraints.

5.7 Benchmarking

i. Comparison of current implementation against best practices in offline-first, nested-data, Android applications.
ii. Reference to standards such as OWASP MASVS (Mobile Application Security Verification Standard) and industry QA frameworks.

6. Duration & Timeline

The entire QA review is expected to be completed within 1 month (4 weeks).

Week 1: Inception & Setup
i. Kick-off meeting, clarification of requirements.
ii. Environment setup and access provisioning (mobile builds, web code, test database).
iii. Review of available documentation.
iv. Drafting of Test Plan and Test Matrix.

Week 2: Code Review & Functional Testing
i. Static code analysis of mobile and web platforms.
ii. Manual and exploratory functional testing of the mobile app.
iii. Initial sync validation (Web ↔ Mobile).
iv. Draft defect log with early findings.

Week 3: Load, Concurrency & Sync Testing
i. Execution of load/concurrency tests with simulated users.
ii. Stress testing of sync workflows (large data volumes, multiple submissions).
iii. Validation of draft-saving, offline-first behaviour, and error handling.
iv. Expansion of defect log with severity classification.

Week 4: Consolidation & Reporting
i. Finalization of QA Review Report.
ii. Preparation of Feature Verification Log, Code Assessment, and Documentation Gap Analysis.
iii. Drafting of remediation recommendations.
iv. Presentation of findings to stakeholders.
v. Handover of all materials (logs, test cases, scripts).

7. Reporting & Communication

i. Weekly progress updates to the designated project manager, summarizing completed tasks, open issues, and risks.
ii. Defect tracking log shared on an ongoing basis, updated daily during Weeks 2–3.
iii. Interim findings shared at the end of Week 2 (midpoint review).
iv. Final report and presentation delivered in Week 4.
v. All communication will be coordinated through a designated single point of contact.

8. Eligibility

The consultant/team must demonstrate:

i. Proven experience in mobile and web application QA and code review.
ii. Strong knowledge of Kotlin and offline-first Android architectures, nested/hierarchical data capture, and sync engines.

iii. Expertise in software architecture, performance optimization, and security standards.
iv. Hands-on experience with load/concurrency testing tools (e.g., JMeter, Locust).
v. Familiarity with modern development frameworks, CI/CD pipelines, and DevOps practices.
vi. Strong analytical, reporting, and presentation skills, with prior experience preparing QA reports for complex systems.

9. Confidentiality

i. All application materials, source code, test data, and findings will be treated as strictly confidential.
ii. The consultant/team shall not disclose, share, or use any information from this engagement outside of the agreed scope.
iii. A non-disclosure agreement (NDA) will be required prior to commencement of work.

10. Application

Interested applicants are requested to prepare and submit a comprehensive proposal covering the following:

a) Technical Proposal

i. A detailed and structured approach to addressing the Scope of Work (Section 3) and delivering the outputs defined under Deliverables (Section 4).
ii. The proposal must explicitly outline the methods, tools, and frameworks that will be used to ensure quality assurance and testing, including code review methodologies, test management approaches, and documentation standards.
iii. Applicants should also indicate additional areas not covered in the ToR that they consider critical for robust QA and system improvement (e.g., security testing, automated regression testing, API stress validation).
iv. A work plan with milestones should be included, showing how progress will be tracked and aligned with the timeline in Section 6.

b) Financial Proposal

i. A clear and itemized costing linked to each deliverable (Section 4), including optional or recommended additional activities.
ii. Applicants must explicitly state the time required to achieve each deliverable, presented in a milestone-based timeline.
iii. The financial proposal must demonstrate the feasibility of the proposed budget and timeline, showing how resources will be allocated effectively.
iv. Any assumptions underlying the costing should be clearly stated.

c) Performance Evaluation

Proposals will be evaluated by GAIN based on the quality and completeness of the Technical Proposal, the feasibility of the Financial Proposal and timeline, and the practicality of the proposed approach to achieving deliverables. GAIN reserves the right to reject any proposal at its sole discretion and without providing an explanation.

d) Submission Instructions

i. The complete proposal (technical and financial proposals, along with company profile) must be submitted by **5 October, 12 AM CET**.
ii. Applications must be sent to **rfp@gainhealth.org** with the subject line: QC-Testing of FortiMapp. Proposals submitted without this subject line will not be considered.
iii. All queries should also be addressed to rfp@gainhealth.org with the subject line: Query regarding QC-Testing of FortiMapp.

11. Terms & Conditions

This section constitutes the full conditions of this RFP. Participation in this process automatically constitutes acceptance of these Terms & Conditions by the Bidder.

i. Jurisdiction
Any contract resulting from this RFP shall be governed by and construed in accordance with the laws of Switzerland. The courts of Geneva, Switzerland shall have exclusive jurisdiction to resolve any disputes arising under or in connection with this RFP or any subsequent contract.

ii. Late Proposals
Any Proposal received after the stated deadline shall not be evaluated and may be rejected at GAIN's absolute discretion. No exceptions will be made under any circumstances.

iii. Disclaimer of Liability
i. This RFP does not constitute an offer to contract. It is a request for Proposals only and does not commit GAIN to award a contract, accept any Proposal, or pay any costs incurred in preparing a Proposal.
ii. Bidders will not be reimbursed or compensated for costs associated with the preparation and submission of Proposals, participation in meetings or presentations, or the provision of additional information.
iii. While the information in this RFP has been prepared in good faith, GAIN makes no representation or warranty as to the completeness or accuracy of the information contained herein. Neither GAIN, its staff, its agents, nor its advisors shall be liable for any errors or omissions, nor for any loss or damage suffered by any Bidder in reliance on this RFP.

iv. Acceptance of Proposals
i. GAIN reserves the right to accept or reject any Proposal, in whole or in part, at its sole discretion and without providing any explanation.
ii. GAIN is under no obligation to accept the lowest-priced Proposal or any Proposal.
iii. A Proposal may be accepted by GAIN for all or part of the services described in this RFP.
iv. No Proposal shall be deemed accepted, in whole or in part, unless and until a formal written contract has been signed by both GAIN and the successful Bidder.

v. Right to Amend or Cancel
i. GAIN reserves the right to amend the RFP, including its requirements and terms, at any time prior to the submission deadline. Any such amendment shall be issued in writing and communicated to all prospective Bidders.

ii. GAIN further reserves the right to cancel, suspend, or terminate the RFP process at any time, without incurring any liability to Bidders.

vi. Validity of Proposals
Proposals must remain valid and binding for a minimum of ninety (90) calendar days following the RFP closing date. During this period, Proposals may not be modified or withdrawn without prior written consent from GAIN.

vii. Withdrawal of Proposals
Bidders may withdraw their Proposals at any time before the RFP closing date by submitting written notice to GAIN. No withdrawal will be accepted after the submission deadline.

viii. Interpretation of Requirements
i. Bidders are responsible for ensuring that they have obtained all necessary information to prepare their Proposals. Failure to do so shall not relieve the Bidder of any obligations if selected.
ii. Bidders must satisfy themselves regarding the accuracy and interpretation of the terminology and requirements set out in the RFP.
iii. By submitting a Proposal, the Bidder warrants that it has fully understood the requirements and obligations associated with the services described.

ix. Assumptions
Any assumptions relied upon by a Bidder in preparing its Proposal must be explicitly stated within the Proposal. GAIN shall not be responsible for any incorrect assumptions made by the Bidder.

x. Confidentiality
i. All information provided by GAIN in connection with this RFP shall be treated as strictly confidential.
ii. Except as required for Proposal preparation, Bidders must not disclose to any third party any part of this RFP or related documents without GAIN's prior written consent.
iii. Bidders must ensure that their employees, agents, and subcontractors are bound by confidentiality obligations no less restrictive than those contained herein.

xi. Feedback to Unsuccessful Bidders
GAIN endeavours to provide general feedback to unsuccessful Bidders upon request. However, GAIN is under no obligation to provide detailed evaluation scores, comparative analysis, or proprietary information relating to the assessment process.

xii. Inconsistencies and Omissions
Bidders must promptly notify GAIN in writing of any errors, omissions, or inconsistencies identified in this RFP. Failure to do so may result in disqualification.

xiii. Collusive Behaviour and Misconduct
i. Bidders must not engage in collusive, anti-competitive, or fraudulent behaviour, including but not limited to:
– fixing or adjusting the amount of their Proposal in agreement with another party.
– sharing or communicating Proposal pricing or related information with another party (except where strictly required to obtain quotations for Proposal preparation).
– entering into arrangements with another party to refrain from submitting a Proposal.

– offering inducements, payments, or rewards to any person in relation to the preparation or acceptance of a Proposal.
ii. Any Bidder engaging in such conduct shall be immediately disqualified and may be subject to civil and/or criminal liability.

xiv. Amendments During Evaluation
GAIN reserves the right to request clarifications or additional information from Bidders during the evaluation process. Failure to respond adequately within the requested timeframe may result in disqualification.