Development of AR System Technology for Quality Information Display

Detailed Project Implementation Plan

Hanoi University of Science and Technology
Institute of Energy Technology

Hwashin Co., Ltd

Project Period: August 2025 - August 2026

Contents

1 Executive Summary

This document presents a comprehensive implementation plan for the development of an Augmented Reality (AR) system designed to enhance quality inspection processes for formed parts in manufacturing environments. The project is a collaboration between Hwashin Co., Ltd (Korea) and the Institute of Energy Technology at Hanoi University of Science and Technology (Vietnam).

1.1 Project Overview

- **Project Title:** Development of AR System Technology for Quality Information Display
- **Total Budget:** $20,000 USD
- **Duration:** August 15, 2025 - August 31, 2026 (12.5 months)
- **Primary Technology:** LiDAR-based AR on iOS devices (iPad/iPhone)
- **Project Manager:** Professor Nguyen Duc Toan (IET)
- **Responsible Manager:** Team Leader Yeo In-Joo (HWASHIN)

1.2 Strategic Importance

Augmented Reality technology offers significant potential to improve the accuracy and efficiency of visual inspection for manufactured parts. This project addresses the critical need for real-time quality assessment by developing an innovative mobile application that leverages LiDAR scanning capabilities to recognize objects and overlay quality data directly onto physical parts.

2 Project Background and Motivation

2.1 Industry Context

In modern manufacturing, particularly in the automotive and metal forming industries, quality inspection of formed parts is a critical yet time-consuming process. Traditional inspection methods involve:

- Manual measurements with calipers and gauges
- Time-intensive coordinate measuring machine (CMM) operations
- Limited real-time feedback to production operators
- Difficulty in visualizing dimensional deviations

2.2 Technology Gap

Current inspection systems lack:

- Mobile, on-site inspection capabilities
- Intuitive visualization of quality deviations
- Real-time comparison with design specifications
- Integration of 3D scanning with quality data mapping

2.3 Solution Approach

This project develops a mobile AR application that:

1. Utilizes iPad/iPhone LiDAR scanners for 3D object recognition
2. Imports template databases containing design specifications
3. Compares scanned geometry with CAD data
4. Visualizes dimensional differences using color-coded heat maps
5. Provides immediate feedback to quality inspectors

3 Project Objectives

3.1 Primary Objective

Develop a fully functional mobile AR application capable of recognizing manufactured parts using LiDAR technology and displaying quality information by comparing scanned objects with design templates.

3.2 Specific Technical Objectives

1. **Object Recognition:** Achieve $> 90\%$ recognition accuracy for targeted objects/parts
2. **Data Import:** Successfully import and display 100% of template databases
3. **Geometric Comparison:** Compare 100% of scanned objects with imported design data
4. **Visualization:** Display dimensional differences as intuitive color-coded overlays
5. **Performance:** Ensure real-time processing suitable for shop floor use

3.3 Deliverable Objectives

- Complete and documented iOS mobile application
- User manual and technical documentation
- Training materials for end users
- Progress and final technical reports
- Tested and validated system with HWASHIN sample data

4 Technical Requirements and Specifications

4.1 Hardware Requirements

4.1.1 Development Devices

- **Primary Platform:** iPad Pro with LiDAR scanner
- **Secondary Platform:** iPhone Pro series with LiDAR capability
- **Minimum iOS Version:** iOS 14.0 or higher
- **Processing:** A12 Bionic chip or newer
- **Storage:** Minimum 64GB for application and database storage

4.1.2 LiDAR Scanner Specifications

- Range: Up to 5 meters
- Point cloud density: Suitable for part-level detail
- Frame rate: 30 Hz minimum
- Depth precision: Millimeter-level accuracy

4.2 Software Requirements

4.2.1 Development Environment

- **IDE:** Xcode 12.0 or later
- **Programming Language:** Swift 5.0+
- **Frameworks:**
  - ARKit 4.0+ (for AR functionality)
  - RealityKit (for 3D rendering)
  - SceneKit (for 3D scene management)
  - CoreML (for machine learning-based recognition)
  - Vision framework (for computer vision tasks)

4.2.2 Core Functionalities

1. **3D Scanning Module**
   - Real-time LiDAR point cloud capture
   - Mesh reconstruction from point cloud data
   - Object boundary detection and isolation
2. **Object Recognition Module**

- Feature extraction from scanned geometry
- Template matching algorithm
- Machine learning-based classification
- Pose estimation and alignment

3. **Database Management Module**
   - Import CAD models (STEP, STL, OBJ formats)
   - Store template geometries and metadata
   - Efficient retrieval and caching mechanisms
   - Support for multiple part variants
4. **Comparison and Analysis Module**
   - Point-to-surface distance calculation
   - Dimensional deviation computation
   - Statistical analysis of quality metrics
   - Tolerance zone evaluation
5. **Visualization Module**
   - Color-coded heat map generation
   - AR overlay rendering on physical object
   - Interactive inspection controls
   - Report generation with screenshots

4.3 Performance Specifications

- **Recognition Time:** < 3 seconds per object
- **Comparison Processing:** < 5 seconds for typical parts
- **Frame Rate:** Maintain > 30 FPS during AR visualization
- **Accuracy:** Dimensional measurements within ± 0.5 mm
- **Recognition Success Rate:** $> 90\%$ for trained objects

5 System Architecture

5.1 Overall Architecture

The system follows a modular architecture with the following layers:

System Architecture Layers

1. **Presentation Layer** (User Interface)
↓
2. **Application Layer** (Business Logic)
↓
3. **Data Layer** (Database and File Management)
↓
4. **Hardware Interface Layer** (LiDAR, Camera, Sensors)

5.2 Module Descriptions

5.2.1 Presentation Layer

- **AR View Controller:** Main interface for scanning and visualization
- **Database Browser:** Interface for selecting and managing templates
- **Settings Panel:** Configuration options and calibration tools
- **Report Viewer:** Display inspection results and history

5.2.2 Application Layer

- **Scan Manager:** Controls LiDAR scanning process
- **Recognition Engine:** Implements object identification algorithms
- **Comparison Engine:** Performs geometric analysis
- **Visualization Engine:** Generates AR overlays and heat maps

5.2.3 Data Layer

- **Template Database:** Stores CAD models and specifications
- **Scan Cache:** Temporary storage for captured point clouds
- **Results Database:** Stores inspection history and results
- **Configuration Manager:** Handles user preferences and settings

5.2.4 Hardware Interface Layer

- **ARKit Interface:** Access to LiDAR and motion tracking
- **Camera Interface:** RGB camera for visual feedback
- **File System Interface:** Storage management

5.3 Data Flow

1. User launches application and selects inspection mode
2. System activates LiDAR scanner and AR session
3. User points device at target object
4. LiDAR captures point cloud data in real-time
5. Recognition engine identifies object type from templates
6. System retrieves corresponding CAD model from database
7. Comparison engine calculates dimensional deviations
8. Visualization engine generates color-coded heat map
9. AR overlay is rendered on physical object through device screen
10. User reviews results and optionally saves report

5.4 Technology Stack

Component	**Technology**
Development Language	Swift 5.0+
UI Framework	SwiftUI / UIKit
AR Framework	ARKit 4.0+
3D Rendering	RealityKit / SceneKit
Machine Learning	CoreML
Computer Vision	Vision Framework
Database	Core Data / SQLite
File Formats	STL, OBJ, STEP, USD
Version Control	Git
Project Management	Jira / Trello

6 Development Phases and Timeline

6.1 Phase 1: Project Initiation and Planning

Duration: August 15 - September 30, 2025 (1.5 months)

6.1.1 Activities

1. **Team Formation and Kickoff**
    - Assemble development team
    - Assign roles and responsibilities
    - Conduct kickoff meeting
    - Establish communication protocols
2. **Requirements Analysis**
    - Detailed analysis of HWASHIN's requirements
    - Study sample parts and quality inspection processes
    - Define use cases and user stories
    - Establish success criteria and KPIs
3. **Technical Planning**
    - Finalize technology stack
    - Design system architecture
    - Develop data models for templates and results
    - Create detailed development roadmap
4. **Environment Setup**
    - Configure development workstations
    - Set up version control repository
    - Establish CI/CD pipeline
    - Prepare testing devices
5. **Risk Assessment**
    - Identify technical risks
    - Develop mitigation strategies
    - Establish contingency plans

6.1.2 Deliverables

- **Work Plan** (Due: October 1, 2025)
  - Project goals and objectives
  - Detailed task breakdown
  - Team member assignments
  - Development schedule with milestones
  - Risk management plan

6.2 Phase 2: Core Technology Development

Duration: October 1 - December 31, 2025 (3 months)

6.2.1 Activities

1. **LiDAR Scanning Module (Weeks 1-4)**
   - Implement ARKit session management
   - Develop point cloud capture functionality
   - Create mesh reconstruction algorithms
   - Implement object isolation and segmentation
   - Test scanning accuracy and performance
2. **Template Database System (Weeks 3-6)**
   - Design database schema
   - Implement CAD file import (STL, OBJ formats)
   - Develop template preprocessing pipeline
   - Create database management interface
   - Test with HWASHIN template samples
3. **Object Recognition Engine (Weeks 5-10)**
   - Implement feature extraction algorithms
   - Develop template matching system
   - Train machine learning models for classification
   - Implement pose estimation and alignment
   - Optimize recognition accuracy and speed
4. **Integration Testing**
   - Test scanning-to-recognition pipeline
   - Validate database integration
   - Performance profiling and optimization

6.2.2 Milestones

- Working LiDAR scanning prototype (November 1)
- Template database functional (November 15)
- Object recognition achieving 80% accuracy (December 31)

6.3 Phase 3: Comparison and Visualization Development

Duration: January 1 - March 31, 2026 (3 months)

6.3.1 Activities

1. **Geometric Comparison Engine (Weeks 11-16)**
    - Implement point-to-surface distance calculation
    - Develop deviation analysis algorithms
    - Create statistical analysis tools
    - Implement tolerance zone evaluation
    - Optimize computation performance
2. **Visualization System (Weeks 15-20)**
    - Design color-coding scheme for deviations
    - Implement heat map generation
    - Develop AR overlay rendering
    - Create interactive inspection controls
    - Implement screenshot and report generation
3. **User Interface Development (Weeks 17-22)**
    - Design intuitive UI/UX
    - Implement main AR view controller
    - Create database browser interface
    - Develop settings and configuration panels
    - Implement help and tutorial system

6.3.2 Deliverables

- **Progress Report** (Due: January 31, 2026)
    - Summary of achievements to date
    - Technical challenges and solutions
    - Updated project schedule
    - Demo video of current functionality

6.3.3 Milestones

- Comparison engine functional (February 15)
- Heat map visualization working (March 1)
- Complete UI implementation (March 31)

6.4 Phase 4: Testing, Validation, and Refinement

Duration: April 1 - June 30, 2026 (3 months)

6.4.1 Activities

1. **System Integration Testing (Weeks 23-26)**
    - End-to-end workflow testing
    - Integration of all modules
    - Performance optimization
    - Bug fixing and refinement
2. **Validation with HWASHIN Data (Weeks 27-30)**
    - Import HWASHIN template databases
    - Test with actual part samples provided by HWASHIN
    - Validate recognition accuracy ($> 90\%$ target)
    - Verify comparison accuracy
    - Validate visualization quality
3. **User Acceptance Testing (Weeks 29-32)**
    - Conduct testing sessions with HWASHIN team
    - Gather user feedback
    - Refine UI/UX based on feedback
    - Optimize for real-world usage scenarios
4. **Documentation (Weeks 31-34)**
    - Write technical documentation
    - Create user manual
    - Develop training materials
    - Prepare installation guide

6.4.2 Milestones

- System integration complete (May 1)
- Validation testing passed (June 15)
- All acceptance criteria met (June 30)

6.5 Phase 5: Finalization and Deployment

Duration: July 1 - August 31, 2026 (2 months)

6.5.1 Activities

1. **Final Refinements (Weeks 35-37)**
    - Address any remaining issues
    - Final performance optimization
    - Code cleanup and refactoring
    - Final testing cycle
2. **Deployment Preparation (Weeks 37-39)**
    - Prepare deployment package
    - Create backup and recovery procedures
    - Finalize all documentation
    - Prepare training sessions
3. **Knowledge Transfer (Weeks 39-40)**
    - Conduct training sessions for HWASHIN team
    - Provide hands-on demonstrations
    - Transfer source code and documentation
    - Establish support procedures
4. **Project Closure (Week 40)**
    - Final project review
    - Lessons learned documentation
    - Handover to HWASHIN
    - Project completion certification

6.5.2 Deliverables

- **Final Report** (Due: July 31, 2026)
    - Complete project achievements
    - Technical challenges and solutions
    - Performance metrics and validation results
    - Recommendations for future enhancements
- **Mobile Application** (Due: August 31, 2026)
    - Fully functional iOS application
    - Source code with comments

- Technical documentation
- User manual
- Training materials
- Installation and setup guide

6.6 Timeline Summary

Phase	**Duration**	**Key Milestone**
Phase 1: Planning	Aug-Sep 2025	Work Plan Delivered
Phase 2: Core Development	Oct-Dec 2025	Recognition Engine Ready
Phase 3: Visualization	Jan-Mar 2026	Progress Report Delivered
Phase 4: Testing	Apr-Jun 2026	Validation Complete
Phase 5: Finalization	Jul-Aug 2026	Application Delivered

7 Team Structure and Resources

7.1 Project Organization

7.1.1 HWASHIN Team

- **Representative:** Mr. Seo-jin Chung (President)
- **Responsible Manager:** Team Leader Yeo In-Joo
- **Technical Liaison:** Quality Assurance Team
- **End Users:** Production and Inspection Staff

7.1.2 IET Team

- **Legal Representative:** Assoc. Prof. Dang Tran Tho (Dean)
- **Project Manager:** Professor Nguyen Duc Toan
- **Development Team:** (To be finalized in Work Plan)
  - Lead iOS Developer
  - AR/Computer Vision Specialist
  - 3D Graphics Developer
  - Machine Learning Engineer
  - UI/UX Designer
  - Quality Assurance Engineer
  - Technical Writer

7.2 Roles and Responsibilities

7.2.1 Project Manager

- Overall project coordination
- Communication with HWASHIN
- Resource allocation and management
- Risk management and mitigation
- Progress tracking and reporting

7.2.2 Lead iOS Developer

- System architecture design
- Core application development
- Code quality and standards
- Technical leadership
- Integration oversight

7.2.3 AR/Computer Vision Specialist

- LiDAR scanning implementation
- Object recognition algorithms
- ARKit integration
- Performance optimization

7.2.4 3D Graphics Developer

- Mesh processing and rendering
- Heat map visualization
- AR overlay implementation
- Graphics optimization

7.2.5 Machine Learning Engineer

- Object classification model development
- Feature extraction algorithms
- Model training and optimization
- CoreML integration

7.2.6 UI/UX Designer

- User interface design
- User experience optimization
- Usability testing
- Design documentation

7.2.7 Quality Assurance Engineer

- Test plan development
- Functional and performance testing
- Bug tracking and verification
- Validation with HWASHIN data

7.2.8 Technical Writer

- Technical documentation
- User manual creation
- Training material development
- Report writing

7.3 Resource Requirements

7.3.1 Hardware

- iPad Pro devices with LiDAR (provided by HWASHIN)
- Development workstations (Mac computers)
- Testing devices (various iOS devices)
- Sample parts from HWASHIN

7.3.2 Software

- Xcode development environment
- CAD software for template preparation
- Version control system (Git)
- Project management tools
- Testing and debugging tools

7.3.3 Facilities

- Development laboratory
- Testing environment
- Meeting and collaboration spaces
- Secure storage for equipment

8 Risk Management

8.1 Technical Risks

Risk	Probability	Impact	Mitigation
LiDAR accuracy insufficient	Medium	High	Early prototyping, alternative sensors
Object recognition accuracy below 90%	Medium	High	Advanced ML models, more training data
Performance issues on older devices	Low	Medium	Optimization, minimum spec requirements
CAD import format compatibility	Medium	Medium	Support multiple formats, conversion tools
AR visualization lag	Medium	High	Graphics optimization, reduced polygon count
Database scalability issues	Low	Medium	Efficient indexing, caching strategies

8.2 Project Management Risks

Risk	Probability	Impact	Mitigation
Team member unavailability	Medium	High	Cross-training, backup resources
Schedule delays	Medium	High	Buffer time, agile methodology
Communication gaps with HWASHIN	Low	Medium	Regular meetings, clear protocols
Scope creep	Medium	High	Strict change control, requirement freeze
Budget overrun	Low	Medium	Careful resource planning, contingency

8.3 External Risks

Risk	Probability	Impact	Mitigation
iOS version incompatibility	Low	High	Target stable iOS versions, testing
Hardware availability delays	Low	Medium	Early procurement, alternative sources
HWASHIN sample delay	Medium	High	Use surrogate samples, early coordination
Force majeure events	Low	High	Remote work capability, flexible schedule

8.4 Risk Monitoring and Control

1. **Weekly Risk Reviews:** Assess risk status in team meetings
2. **Risk Register:** Maintain updated risk documentation
3. **Escalation Protocol:** Clear procedures for critical risks
4. **Contingency Planning:** Detailed plans for high-impact risks
5. **Communication:** Transparent reporting to HWASHIN

9 Quality Assurance and Testing

9.1 Quality Objectives

- Zero critical bugs in production release
- $> 90\%$ object recognition accuracy
- < 3 seconds average recognition time
- > 30 FPS AR visualization performance
- ± 0.5 mm dimensional measurement accuracy
- $> 95\%$ user satisfaction rating

9.2 Testing Strategy

9.2.1 Unit Testing

- Test individual modules and functions
- Achieve $> 80\%$ code coverage
- Automated unit tests with XCTest framework
- Continuous integration testing

9.2.2 Integration Testing

- Test interactions between modules
- Validate data flow across system
- Test API and database integration
- Verify ARKit and LiDAR integration

9.2.3 Functional Testing

- Verify all specified requirements
- Test complete user workflows
- Validate against acceptance criteria
- Test with various part types and sizes

9.2.4 Performance Testing

- Measure recognition and processing times
- Test frame rate under various conditions
- Evaluate memory usage and battery consumption
- Stress testing with large databases

9.2.5 Accuracy Validation

- Compare measurements with CMM data
- Test recognition accuracy with diverse samples
- Validate geometric comparison precision
- Test under various lighting conditions

9.2.6 Usability Testing

- User interface intuitiveness
- Learning curve assessment
- Workflow efficiency evaluation
- Feedback collection from HWASHIN users

9.2.7 Acceptance Testing

- Validation with HWASHIN template database
- Testing with actual HWASHIN parts
- Verification of all contractual requirements
- Sign-off by HWASHIN team

9.3 Test Environment

- Development testing: IET laboratory
- Integration testing: Controlled environment with sample parts
- Field testing: HWASHIN production facility (if applicable)
- Acceptance testing: HWASHIN designated location

9.4 Defect Management

1. **Bug Tracking System:** Use Jira or similar tool
2. **Severity Classification:**
    - Critical: Application crash, data loss
    - High: Major functionality broken
    - Medium: Minor functionality issues
    - Low: Cosmetic issues
3. **Resolution Priorities:**
    - Critical: Immediate fix required

- High: Fix within 48 hours
- Medium: Fix within 1 week
- Low: Fix in next release

4. **Verification:** All fixes must be tested and verified
5. **Regression Testing:** Ensure fixes don't introduce new issues

9.5 Quality Metrics

Metric	Target	Measurement Method
Code Coverage	$> 80\%$	XCTest coverage report
Recognition Accuracy	$> 90\%$	Test with 100+ samples
Processing Time	< 3 sec	Performance profiler
Frame Rate	> 30 FPS	ARKit metrics
Bug Density	< 2 per KLOC	Defect tracking system
User Satisfaction	$> 95\%$	Post-deployment survey

10 Budget Allocation and Financial Management

10.1 Total Budget

Total project budget: **$20,000 USD** (including VAT)

10.2 Budget Allocation

Category	Amount (USD)	Percentage
Personnel Costs	$12,000	60%
Equipment & Hardware	$0*	0%
Software & Tools	$1,500	7.5%
Research & Development	$3,000	15%
Testing & Validation	$1,500	7.5%
Documentation & Training	$1,000	5%
Contingency Reserve	$1,000	5%
Total	**$20,000**	**100%**

Table 4: *Hardware devices provided separately by HWASHIN

10.3 Personnel Cost Breakdown

Role	Person-Months	Rate	Total
Project Manager	12	$600	$7,200
Lead Developer	10	$500	$5,000
AR Specialist	8	$450	$3,600
Other Team Members	-	-	$8,200
Subtotal			**$12,000**

10.4 Software and Tools

- Development tools and IDEs: $500
- CAD software licenses: $400
- Project management tools: $200
- Testing and debugging tools: $300
- Other utilities: $100
- **Subtotal: $1,500**

10.5 Research and Development

- Algorithm development and optimization: $1,500
- Machine learning model training: $800
- Prototype development: $400
- Technical research and literature: $300
- **Subtotal: $3,000**

10.6 Testing and Validation

- Test sample preparation: $500
- Validation equipment and materials: $400
- User acceptance testing: $400
- Performance benchmarking: $200
- **Subtotal: $1,500**

10.7 Documentation and Training

- Technical documentation: $400
- User manual development: $300
- Training material creation: $200
- Video tutorials: $100
- **Subtotal: $1,000**

10.8 Payment Schedule

Milestone	Date	Amount	Percentage
Contract Signing	September 2025	$14,000	70%
Project Completion	September 2026	$6,000	30%
Total		**$20,000**	**100%**

10.9 Financial Management

1. **Budget Tracking:** Monthly financial reports
2. **Cost Control:** Regular review of expenditures
3. **Variance Analysis:** Track deviations from budget
4. **Approval Process:** All major expenses require approval
5. **Contingency Use:** Reserve fund for unexpected costs
6. **Transparency:** Financial reports shared with HWASHIN

10.10 HWASHIN Contributions

In addition to the budget, HWASHIN provides:

- iPad/iPhone devices with LiDAR (October 2025)
- Template database samples (November 2025)
- Physical part samples for testing (December 2025)
- Technical consultation and feedback
- Validation and acceptance testing facilities

11 Success Criteria and Acceptance

11.1 Technical Success Criteria

1. **Object Recognition**
   - Recognition accuracy: $> 90\%$ on test samples
   - Recognition time: < 3 seconds per object
   - Robust to various lighting and viewing angles
   - Successful pose estimation and alignment
2. **Data Import**
   - Successfully import 100% of HWASHIN template databases
   - Support for standard CAD formats (STL, OBJ, STEP)
   - Proper handling of metadata and specifications
   - Efficient database retrieval (< 1 second)
3. **Geometric Comparison**
   - Compare 100% of scanned objects with templates
   - Dimensional accuracy: ± 0.5 mm
   - Complete coverage of part surfaces
   - Accurate deviation calculation
4. **Visualization**
   - Clear, intuitive color-coded heat maps
   - Real-time AR overlay (> 30 FPS)
   - Accurate alignment with physical parts
   - Interactive inspection capabilities
5. **Performance**
   - Total inspection time: < 10 seconds
   - Smooth user experience with no lag
   - Efficient battery consumption
   - Stable operation for continuous use

11.2 Functional Success Criteria

1. All features specified in Annex I implemented
2. Complete end-to-end workflow functional
3. Intuitive user interface requiring minimal training
4. Robust error handling and user feedback
5. Data persistence and report generation
6. Help and documentation accessible within app

11.3 Quality Success Criteria

1. Zero critical or high-severity bugs in release
2. Code quality meets industry standards
3. Comprehensive technical documentation
4. Complete user manual and training materials
5. Successful validation with HWASHIN data
6. Positive feedback from HWASHIN users

11.4 Acceptance Process

1. **Deliverable Submission**
   - IET submits completed application and documentation
   - Includes source code, user manual, and technical docs
   - Submission by deadline: August 31, 2026
2. **HWASHIN Review** (1 month)
   - Functional testing against requirements
   - Performance and accuracy validation
   - User acceptance testing
   - Documentation review
3. **Feedback and Modifications** (if needed)
   - HWASHIN provides detailed feedback
   - IET addresses issues within 1 month
   - Re-submission for final review
4. **Certification of Work Completion**
   - HWASHIN issues formal certification
   - Confirms all requirements met
   - Triggers final payment release

11.5 Verification Methods

Criterion	Verification Method	Pass Threshold
Recognition Accuracy	Test with 100+ samples	$> 90\%$ success rate
Processing Speed	Timed measurements	< 3 sec recognition
Import Capability	Import all templates	100% success
Comparison Coverage	Visual inspection	100% surface coverage
Visualization Quality	User evaluation	$> 4/5$ rating
Performance	FPS monitoring	> 30 FPS
Accuracy	CMM comparison	± 0.5 mm
Usability	User survey	$> 4/5$ rating

12 Communication and Reporting

12.1 Communication Plan

12.1.1 Regular Communications

- **Weekly Team Meetings:** Internal IET team coordination
- **Bi-weekly Progress Updates:** Email updates to HWASHIN
- **Monthly Video Conferences:** Detailed progress review with HWASHIN
- **Quarterly In-Person Meetings:** Major milestone reviews (if feasible)

12.1.2 Ad-Hoc Communications

- Technical questions and clarifications: Email or messaging
- Urgent issues: Phone or video call
- Demonstration requests: Scheduled video demos
- Change requests: Formal written proposals

12.2 Reporting Schedule

Report Type	Frequency	Contents
Progress Update	Bi-weekly	Brief status, achievements, next steps
Monthly Report	Monthly	Detailed progress, issues, schedule status
Work Plan	Once	Complete project plan and schedule
Progress Report	Mid-project	Comprehensive progress and technical details
Final Report	Once	Complete project summary and outcomes

12.3 Formal Deliverable Reports

12.3.1 Work Plan (October 1, 2025)

- Project goals and objectives
- Detailed task breakdown with schedule
- Team composition and roles
- Resource allocation
- Risk management plan
- Quality assurance plan

12.3.2 Progress Report (January 31, 2026)

- Achievements to date
- Technical milestones completed
- Challenges encountered and solutions
- Updated project schedule
- Demo video of current functionality
- Risks and mitigation status

12.3.3 Final Report (July 31, 2026)

- Complete project summary
- Technical achievements and innovations
- Challenges and problem-solving approach
- Performance metrics and validation results
- Lessons learned
- Recommendations for future enhancements
- Acknowledgments

12.4 Documentation Deliverables

1. **Technical Documentation**
    - System architecture description
    - Module specifications
    - API documentation
    - Database schema

- Algorithm descriptions
- Code comments and README files

2. **User Manual**
   - Installation and setup guide
   - User interface overview
   - Step-by-step workflows
   - Feature descriptions
   - Troubleshooting guide
   - FAQ section
3. **Training Materials**
   - Quick start guide
   - Video tutorials
   - Training presentation slides
   - Practice exercises
   - Best practices guide
4. **Administrative Documents**
   - Meeting minutes
   - Change request logs
   - Test reports
   - Acceptance certificates

12.5 Issue Escalation

1. **Level 1:** Team member identifies issue
2. **Level 2:** Project manager attempts resolution
3. **Level 3:** Dean and HWASHIN manager involved
4. **Level 4:** Executive leadership from both parties

Escalation Triggers:

- Critical technical blocker
- Major schedule delay risk
- Scope change request
- Budget overrun threat
- Force majeure event

13 Intellectual Property and Confidentiality

13.1 Intellectual Property Rights

As stipulated in Article 4 of the Implementing Agreement:

1. **Ownership:** All deliverables and results are under HWASHIN's sole ownership and intellectual property rights
2. **Use Rights:** Only HWASHIN has the right to use deliverables for further development
3. **Patent Rights:** If HWASHIN pursues patent applications:
   - IET's project manager must be notified in advance
   - HWASHIN covers all patent-related expenses
4. **Assignment Restrictions:** Assignment to third parties requires HWASHIN's consent

13.2 Confidentiality Obligations

13.2.1 Confidential Information

Includes but not limited to:

- Agreement terms and conditions
- Technical specifications and designs
- CAD models and template databases
- Quality data and inspection results
- Source code and algorithms
- Business processes and strategies
- Financial information

13.2.2 Handling Procedures

1. Limit access to authorized team members only
2. Use secure storage and transmission methods
3. Mark documents appropriately as confidential
4. Implement non-disclosure agreements with all staff
5. Secure disposal of confidential materials when no longer needed

13.2.3 Academic Publications

- Academic papers and conference presentations require HWASHIN's prior consent
- Approval process to be established early in project
- Appropriate acknowledgment of both parties in publications

13.3 Data Security

1. **Access Control:** Role-based access to repositories and systems
2. **Encryption:** Encrypt sensitive data in storage and transmission
3. **Backups:** Regular encrypted backups of all project data
4. **Version Control:** Secure Git repository with access logs
5. **Device Security:** Password protection and encryption on all devices
6. **Network Security:** VPN and firewall protection

14 Future Enhancements and Scalability

14.1 Potential Future Features

14.1.1 Short-term Enhancements (Next 6-12 months)

1. **Multi-part Inspection:** Scan and compare multiple parts simultaneously
2. **Batch Processing:** Queue multiple inspections for efficient workflows
3. **Historical Tracking:** Track part quality trends over time
4. **Cloud Sync:** Synchronize data across multiple devices
5. **Advanced Reporting:** Statistical analysis and custom report generation

14.1.2 Medium-term Enhancements (1-2 years)

1. **AI-Powered Defect Classification:** Automatic categorization of defect types
2. **Predictive Quality:** Machine learning to predict quality issues
3. **Integration with ERP/MES:** Connect with enterprise systems
4. **Multi-language Support:** Localization for different regions
5. **Collaborative Features:** Real-time sharing and annotations

14.1.3 Long-term Vision (2+ years)

1. **Industry 4.0 Integration:** Full smart factory integration
2. **Digital Twin:** Real-time synchronization with digital models
3. **Automated Corrective Actions:** AI-driven process adjustments
4. **Cross-platform Support:** Android and web-based versions
5. **Advanced AR:** Holographic displays and wearable AR devices

14.2 Scalability Considerations

14.2.1 Technical Scalability

- **Database:** Design for large template libraries (1000+ parts)
- **Performance:** Optimize algorithms for complex geometries
- **Modularity:** Component-based architecture for easy extensions
- **API Design:** RESTful APIs for future integrations

14.2.2 Business Scalability

- **Multi-tenant:** Support for multiple companies/facilities
- **Licensing:** Flexible licensing models for different use cases
- **Deployment:** Easy distribution through App Store or MDM
- **Support:** Scalable support and maintenance structure

14.3 Technology Roadmap

1. **2026-2027:** Refinement and optimization based on field use
2. **2027-2028:** Advanced AI features and cloud integration
3. **2028-2029:** Enterprise-wide deployment capabilities
4. **2029+:** Next-generation AR hardware and Industry 4.0 integration

15 Sustainability and Maintenance

15.1 Post-Delivery Support

15.1.1 Warranty Period

- Duration: 3 months after delivery (September - November 2026)
- Coverage: Bug fixes and critical issues
- Response time: Within 48 hours for critical issues
- Updates: Patches and minor updates as needed

15.1.2 Extended Support (Optional)

- Extended support contracts available beyond warranty
- Includes: Updates, bug fixes, technical consultation
- Terms: To be negotiated separately if desired

15.2 Maintenance Guidelines

15.2.1 For HWASHIN

1. **Regular Updates:** Install iOS and app updates promptly
2. **Device Maintenance:** Keep devices clean and calibrated
3. **Backup:** Regular backups of template databases
4. **User Training:** Ongoing training for new users
5. **Feedback Collection:** Gather user feedback for improvements

15.2.2 For Future Developers

1. **Code Quality:** Maintain coding standards and documentation
2. **Testing:** Comprehensive tests before any updates
3. **Version Control:** Proper Git workflow and branching
4. **Compatibility:** Test with new iOS versions
5. **Performance Monitoring:** Track app performance metrics

15.3 Knowledge Transfer

1. **Code Handover:** Complete source code with documentation
2. **Training Sessions:** Hands-on training for HWASHIN technical staff
3. **Documentation:** Comprehensive technical and user documentation
4. **Support Channel:** Establish communication channel for questions
5. **Best Practices:** Document operational best practices

16 Conclusion

16.1 Project Summary

This detailed implementation plan outlines a comprehensive approach to developing an innovative Augmented Reality system for quality inspection in manufacturing. The project leverages cutting-edge LiDAR technology on iOS devices to provide real-time, intuitive visualization of dimensional deviations in formed parts.

16.2 Key Success Factors

1. **Strong Partnership:** Collaboration between HWASHIN and IET
2. **Clear Requirements:** Well-defined objectives and acceptance criteria
3. **Experienced Team:** Skilled developers and specialists
4. **Agile Methodology:** Flexible approach with regular feedback
5. **Quality Focus:** Rigorous testing and validation
6. **Risk Management:** Proactive identification and mitigation
7. **Effective Communication:** Regular updates and transparency

16.3 Expected Outcomes

16.3.1 Technical Outcomes

- Fully functional mobile AR application
- High-accuracy object recognition system
- Intuitive quality visualization interface
- Robust and performant implementation
- Comprehensive documentation

16.3.2 Business Outcomes

- Improved quality inspection efficiency
- Reduced inspection time and costs
- Enhanced quality control capabilities
- Foundation for future AR applications
- Competitive advantage in manufacturing

16.3.3 Research Outcomes

- Advanced AR application development expertise
- Novel approaches to manufacturing quality inspection
- Potential for academic publications
- Foundation for future research projects

16.4 Commitment

Both HWASHIN and IET are committed to the successful execution of this project. Through dedicated effort, technical excellence, and close collaboration, we will deliver an innovative solution that advances the state of quality inspection technology in manufacturing.

This detailed plan serves as a roadmap for the successful development and delivery of the AR Quality Information Display System. It will be refined and updated as the project progresses to reflect actual progress and any necessary adjustments.

A Appendix A: Glossary of Terms

AR (Augmented Reality): Technology that overlays digital information onto the real world

LiDAR (Light Detection and Ranging): Laser-based 3D scanning technology

CAD (Computer-Aided Design): Software for creating digital 3D models

Point Cloud: Collection of 3D points representing a scanned surface

Mesh: 3D surface model composed of vertices, edges, and faces

Heat Map: Color-coded visualization showing variations in values

Template Database: Collection of reference CAD models and specifications

Dimensional Deviation: Difference between actual and design dimensions

CMM (Coordinate Measuring Machine): Precision measurement device

ARKit: Apple's framework for AR development on iOS

CoreML: Apple's machine learning framework

FPS (Frames Per Second): Measurement of rendering performance

B Appendix B: Reference Documents

1. Implementing Agreement dated August 15, 2025
2. Annex I: Deliverables and Assessment Criteria
3. ARKit Documentation (Apple Developer)
4. LiDAR Scanner Technical Specifications
5. Quality Inspection Standards (ISO 9001, etc.)
6. Swift Programming Language Guide
7. iOS Human Interface Guidelines

C Appendix C: Contact Information

C.1 HWASHIN Co., Ltd

- Address: 14, Eonhagongdan 1-gil, Yeongcheon-si, Gyeongsangbuk-do, 38828, Korea
- Tel: +82-54-330-5430
- Fax: +82-54-335-0176
- Representative: Mr. Seo-jin Chung (President)
- Responsible Manager: Team Leader Yeo In-Joo

C.2 Institute of Energy Technology (IET)

- Address: No. 1, Dai Co Viet Street, Bach Mai, Ha Noi, Vietnam
- Tel: +84 24 38 68 2625
- Legal Representative: Assoc. Prof. Dang Tran Tho (Dean)
- Project Manager: Professor Nguyen Duc Toan