Deployment Guide: Smart Waste Management DePIN

Pre-Deployment Planning

Site Assessment Checklist

Location Analysis

```
Physical Requirements:
├── Bin Accessibility: 2m clearance minimum
├── Solar Exposure: 4+ hours direct sunlight
├── Network Coverage: WiFi or cellular signal
├── Security Level: Vandalism risk assessment
├── Maintenance Access: Service vehicle clearance
└── Regulatory Compliance: Local permits required

Environmental Factors:
├── Temperature Range: -20°C to +60°C operation
├── Precipitation: IP65 rating sufficient
├── Wind Load: Mounting structure adequacy
├── Foot Traffic: High/medium/low classification
├── Contamination Risk: Chemical exposure potential
└── Wildlife Interference: Animal-proofing needs
```

Stakeholder Engagement

```
Municipal Partners:
├── Waste Management Department
├── IT/Technology Department
├── Public Works Division
├── Environmental Services
├── Legal/Procurement Team
└── Community Relations

Community Outreach:
├── Neighborhood Associations
├── Environmental Groups
├── Schools and Universities
├── Local Businesses
├── Senior Centers
└── Youth Organizations
```

Regulatory Requirements

Permits & Approvals

- **Installation Permits**: Public space modifications
- **Electrical Permits**: Solar panel installations
- **Telecommunications**: Radio frequency approvals
- **Data Privacy**: GDPR/CCPA compliance documentation
- **Environmental**: Impact assessment if required
- **Insurance**: Liability coverage for public installations

Compliance Standards

- **FCC Part 15**: Radio emissions certification
- **UL Listed**: Electrical safety standards
- **IP65 Rating**: Ingress protection verification
- **RoHS Compliance**: Hazardous substance restrictions
- **WEEE Directive**: End-of-life recycling plan

Hardware Deployment

Phase 1: Pilot Installation (25 Nodes)

Timeline: 6-8 Weeks

```
Week 1-2: Site Preparation
├── Final site surveys and measurements
├── Permit applications and approvals
├── Hardware procurement and testing
├── Installation team training
└── Community notification campaign

Week 3-4: Infrastructure Setup
├── Mounting hardware installation
├── Electrical connections (if required)
├── Network connectivity testing
├── Initial sensor calibration
└── Security system activation

Week 5-6: Node Deployment
├── Hardware installation and configuration
├── Software deployment and testing
├── Network connectivity verification
├── User interface setup
└── Initial data collection validation

Week 7-8: Testing & Optimization
├── System integration testing
├── User acceptance testing
├── Performance optimization
├── Documentation completion
└── Go-live preparation
```

Installation Process (Per Node)

Type A Node (Basic Fill Monitoring)

```
Step 1: Mounting Preparation (30 minutes)
├── Mark mounting holes on bin/post
├── Drill pilot holes (M6 bolts)
├── Install mounting bracket
├── Apply thread locker to bolts
└── Verify secure attachment

Step 2: Hardware Installation (20 minutes)
├── Mount enclosure to bracket
├── Install ultrasonic sensor
├── Connect battery pack
├── Attach solar panel
└── Seal all connections

Step 3: Configuration (15 minutes)
├── Power on and initial boot
├── WiFi network configuration
├── Sensor calibration
├── Location GPS coordinates
└── Initial data transmission test

Step 4: Testing & Validation (10 minutes)
├── Fill level measurement test
├── Data transmission verification
├── Battery charging confirmation
├── Tamper detection test
└── Final system check
```

Type B Node (Advanced AI-Powered)

```
Step 1: Mounting & Power (45 minutes)
├── Install reinforced mounting plate
├── Mount solar panel and controller
├── Install battery compartment
├── Run power cables through conduit
└── Ground system properly

Step 2: Sensor Installation (30 minutes)
├── Mount camera module with weather shield
├── Install load cell platform
├── Position ultrasonic sensor array
├── Connect LoRaWAN antenna
└── Verify all sensor connections

Step 3: System Configuration (25 minutes)
```

```
├── Raspberry Pi initial setup
├── Camera calibration and focus
├── Load cell zero calibration
├── Network connectivity setup
└── AI model deployment

Step 4: Integration Testing (20 minutes)
├── End-to-end data flow test
├── AI classification accuracy
├── Real-time dashboard update
├── Alert system verification
└── Performance baseline establishment
```

Phase 2: Municipal Rollout (200 Nodes)

Deployment Strategy

```
Geographic Distribution:
├── High-Traffic Areas: 40% (80 nodes)
│   ├── Downtown commercial districts
│   ├── Transit stations and hubs
│   ├── Shopping centers and malls
│   └── Tourist attractions
├── Residential Areas: 35% (70 nodes)
│   ├── Apartment complexes
│   ├── Suburban neighborhoods
│   ├── Senior living communities
│   └── Student housing areas
├── Institutional: 15% (30 nodes)
│   ├── Schools and universities
│   ├── Government buildings
│   ├── Hospitals and clinics
│   └── Community centers
└── Industrial/Commercial: 10% (20 nodes)
    ├── Office parks
    ├── Manufacturing facilities
    ├── Warehouses and distribution
    └── Construction sites
```

Installation Teams

```
Team Structure (per 5-node crew):
├── Lead Technician: Installation oversight
├── Hardware Specialist: Sensor configuration
├── Network Engineer: Connectivity setup
├── Safety Officer: Site safety compliance
└── Documentation: Installation records
```

```
Daily Capacity:
├── Type A Nodes: 8-10 installations per day
├── Type B Nodes: 4-6 installations per day
├── Travel Time: 20% of schedule
├── Testing/Validation: 30% of schedule
└── Documentation: 10% of schedule

Quality Assurance:
├── 100% functional testing
├── 24-hour burn-in period
├── Random audit inspections
├── Customer acceptance sign-off
└── Warranty registration
```

Software Deployment

Blockchain Infrastructure

Smart Contract Deployment

```
# Deploy to Solana Devnet (Testing)
solana config set --url https://api.devnet.solana.com
solana program deploy target/deploy/waste_management.so

# Verify deployment
solana program show <PROGRAM_ID>

# Initialize program state
solana program invoke <PROGRAM_ID> initialize_program

# Deploy to Mainnet (Production)
solana config set --url https://api.mainnet-beta.solana.com
solana program deploy target/deploy/waste_management.so --upgrade-authority
<AUTHORITY_KEYPAIR>
```

Token Creation & Distribution

```
# Create WASTE token
spl-token create-token --decimals 6

# Create token accounts
spl-token create-account <TOKEN_MINT>

# Initial token minting
spl-token mint <TOKEN_MINT> 1000000000 <DESTINATION_ACCOUNT>

# Set up distribution wallets
spl-token create-account <TOKEN_MINT> --owner <COMMUNITY_WALLET>
```

```
spl-token create-account <TOKEN_MINT> --owner <DEVELOPMENT_WALLET>
spl-token create-account <TOKEN_MINT> --owner <TREASURY_WALLET>
```

Node Software Configuration

Basic Node (Type A) Setup

```
# Flash firmware to ESP32
esptool.py --chip esp32s3 --port COM3 write_flash 0x0 waste_node_basic.bin

# Configure WiFi credentials
waste-config --wifi-ssid "MunicipalNetwork" --wifi-password "SecurePass123"

# Set location and node ID
waste-config --location "40.7128,-74.0060" --node-id "NYC_001_BASIC"

# Configure reporting intervals
waste-config --report-interval 300 --heartbeat-interval 3600

# Start monitoring service
systemctl enable waste-node-basic
systemctl start waste-node-basic
```

Advanced Node (Type B) Setup

```
# Raspberry Pi OS installation and updates
sudo apt update && sudo apt upgrade -y
sudo apt install python3-pip docker.io -y

# Install waste management software
git clone https://github.com/waste-depin/node-software.git
cd node-software
pip3 install -r requirements.txt

# Configure AI models
python3 setup_ai_models.py --download-models --optimize-for-pi4

# Camera calibration
python3 calibrate_camera.py --interactive

# Load cell calibration
python3 calibrate_loadcell.py --known-weight 10.0

# Network configuration
sudo nano /etc/waste-node/config.yaml
# Edit: location, node_id, network_settings, ai_settings

# Start services
```

```
sudo systemctl enable waste-node-advanced
sudo systemctl start waste-node-advanced
```

Dashboard & API Deployment

Backend Services (Docker)

```
# docker-compose.yml
version: '3.8'
services:
  api-server:
    image: waste-depin/api-server:latest
    ports:
      - "8080:8080"
    environment:
      - DATABASE_URL=postgresql://user:pass@db:5432/waste_db
      - SOLANA_RPC_URL=https://api.mainnet-beta.solana.com
      - JWT_SECRET=your-jwt-secret
    depends_on:
      - database
      - redis

  database:
    image: postgres:14
    environment:
      - POSTGRES_DB=waste_db
      - POSTGRES_USER=waste_user
      - POSTGRES_PASSWORD=secure_password
    volumes:
      - postgres_data:/var/lib/postgresql/data

  redis:
    image: redis:7-alpine
    ports:
      - "6379:6379"

  frontend:
    image: waste-depin/dashboard:latest
    ports:
      - "3000:3000"
    environment:
      - REACT_APP_API_URL=http://localhost:8080
      - REACT_APP_SOLANA_NETWORK=mainnet-beta

volumes:
  postgres_data:
```

Deployment Commands

```
# Deploy to production server
docker-compose -f docker-compose.prod.yml up -d

# Initialize database
docker exec waste-api python manage.py migrate
docker exec waste-api python manage.py create_admin_user

# Configure reverse proxy (nginx)
sudo nano /etc/nginx/sites-available/waste-dashboard
sudo ln -s /etc/nginx/sites-available/waste-dashboard /etc/nginx/sites-enabled/
sudo systemctl reload nginx

# SSL certificate setup
sudo certbot --nginx -d dashboard.waste-depin.org
```

Network Configuration

Connectivity Options

WiFi Network Setup

```
Municipal WiFi Requirements:
├── SSID: WasteManagement_IoT
├── Security: WPA3-Enterprise preferred
├── Bandwidth: 1 Mbps minimum per node
├── Range: 100m coverage radius
├── Backup: Cellular failover capability
└── VLAN: Isolated IoT network segment

Configuration Steps:
1. Create dedicated IoT VLAN
2. Configure RADIUS authentication
3. Set bandwidth limits per device
4. Enable network monitoring
5. Configure firewall rules
6. Test connectivity and failover
```

LoRaWAN Gateway Deployment

```
Gateway Specifications:
├── Coverage: 2-5km urban, 15km rural
├── Capacity: 1000+ nodes per gateway
├── Backhaul: Ethernet or cellular
├── Power: PoE+ or AC adapter
├── Mounting: Rooftop or tower
└── Cost: $300-800 per gateway
```

```
Deployment Locations:
├── Municipal buildings (city hall, libraries)
├── Water towers and communication towers
├── Schools and community centers
├── Fire stations and police stations
└── Public works facilities
```

Data Flow Architecture

Real-time Data Pipeline

```
Node Sensors → Edge Processing → Gateway → Cloud API → Dashboard
     ↓              ↓              ↓         ↓         ↓
  Raw Data    Filtered Data   Compressed  Processed  Visualized
  (1Hz)       (0.1Hz)        (Batch)     (Real-time) (Web UI)
```

Blockchain Integration

```
Smart Contract Events:
├── Node Registration: New hardware deployment
├── Data Verification: Sensor reading validation
├── Reward Distribution: Token allocation
├── Governance Votes: Community decisions
└── System Updates: Protocol upgrades

Transaction Flow:
1. Sensor data aggregated locally
2. Merkle proof generated for batch
3. Submit proof to smart contract
4. Validate data integrity on-chain
5. Trigger reward distribution
6. Update user balances
```

Testing & Validation

System Integration Testing

Functional Tests

```
Hardware Validation:
├── Sensor accuracy: ±5% tolerance
├── Battery life: >6 months Type A, >3 months Type B
├── Solar charging: Full charge in 8 hours sunlight
├── Network connectivity: 99%+ uptime target
├── Tamper detection: 100% alert rate
└── Environmental resistance: IP65 compliance
```

```
Software Validation:
├── Data transmission: <1% packet loss
├── API response time: <200ms average
├── Dashboard load time: <3 seconds
├── Mobile app performance: 60fps UI
├── Blockchain sync: <30 second confirmation
└── AI accuracy: >90% waste classification
```

Performance Benchmarks

```
Scalability Testing:
├── 1,000 nodes: Baseline performance
├── 10,000 nodes: Load testing
├── 100,000 nodes: Stress testing
├── 1,000,000 nodes: Ultimate capacity
└── Failure scenarios: Graceful degradation

Load Testing Results:
├── API throughput: 10,000 requests/second
├── Database queries: <50ms average
├── Blockchain TPS: 2,000 transactions/second
├── Dashboard users: 50,000 concurrent
└── Mobile app users: 100,000 concurrent
```

User Acceptance Testing

Community Beta Program

```
Beta Participants:
├── 100 households across 5 neighborhoods
├── 20 municipal staff members
├── 10 waste management professionals
├── 15 environmental advocates
└── 5 technology enthusiasts

Testing Duration: 8 weeks
├── Week 1-2: Onboarding and training
├── Week 3-4: Basic functionality testing
├── Week 5-6: Advanced features testing
├── Week 7-8: Feedback collection and refinement

Success Criteria:
├── 80%+ user satisfaction rating
├── 90%+ system reliability
├── 95%+ data accuracy
├── <5% support ticket rate
└── 70%+ daily active usage
```

Maintenance & Support

Preventive Maintenance Schedule

Monthly Tasks

- Visual inspection of all nodes
- Battery voltage and charging verification
- Solar panel cleaning and inspection
- Network connectivity testing
- Software update deployment
- Data backup verification

Quarterly Tasks

- Comprehensive system health check
- Sensor calibration verification
- Hardware component replacement (wear items)
- Security audit and penetration testing
- Performance optimization review
- User feedback analysis and implementation

Annual Tasks

- Complete hardware refresh assessment
- Major software version upgrades
- Regulatory compliance review
- Insurance and warranty renewals
- Expansion planning and budgeting
- Technology roadmap updates

Support Infrastructure

Technical Support Tiers

```
Tier 1: Community Support
├── Online documentation and FAQs
├── Community forum and chat
├── Video tutorials and guides
├── Mobile app help system
└── Automated troubleshooting

Tier 2: Professional Support
├── Email support (24-hour response)
├── Phone support (business hours)
├── Remote diagnostics and repair
├── On-site support (if required)
└── Training and consultation
```

```
Tier 3: Enterprise Support
├── Dedicated account manager
├── 24/7 emergency support
├── Custom integration assistance
├── Priority feature development
└── SLA guarantees (99.9% uptime)
```

Monitoring & Alerting

```
System Monitoring:
├── Node health and connectivity
├── Battery levels and charging status
├── Data transmission quality
├── API performance metrics
├── Blockchain synchronization
└── User engagement analytics

Alert Thresholds:
├── Node offline >15 minutes: Warning
├── Battery <20%: Maintenance required
├── Data accuracy <90%: Investigation
├── API response >500ms: Performance issue
├── User complaints >5/day: Priority review
└── Security events: Immediate response
```

Comprehensive deployment ensuring reliable, scalable, and maintainable waste management infrastructure.