Real-Time GPS-Free Underground Worker Tracking System

ESP32, BLE, IMU & Django Implementation

Technical Implementation Guide For Mining & Cave Operations

Critical Problem: Worker Safety Underground

GPS Blackout Zones

Underground mines and caves completely block GPS signals, leaving workers invisible to surface monitoring systems.

Cost Barriers

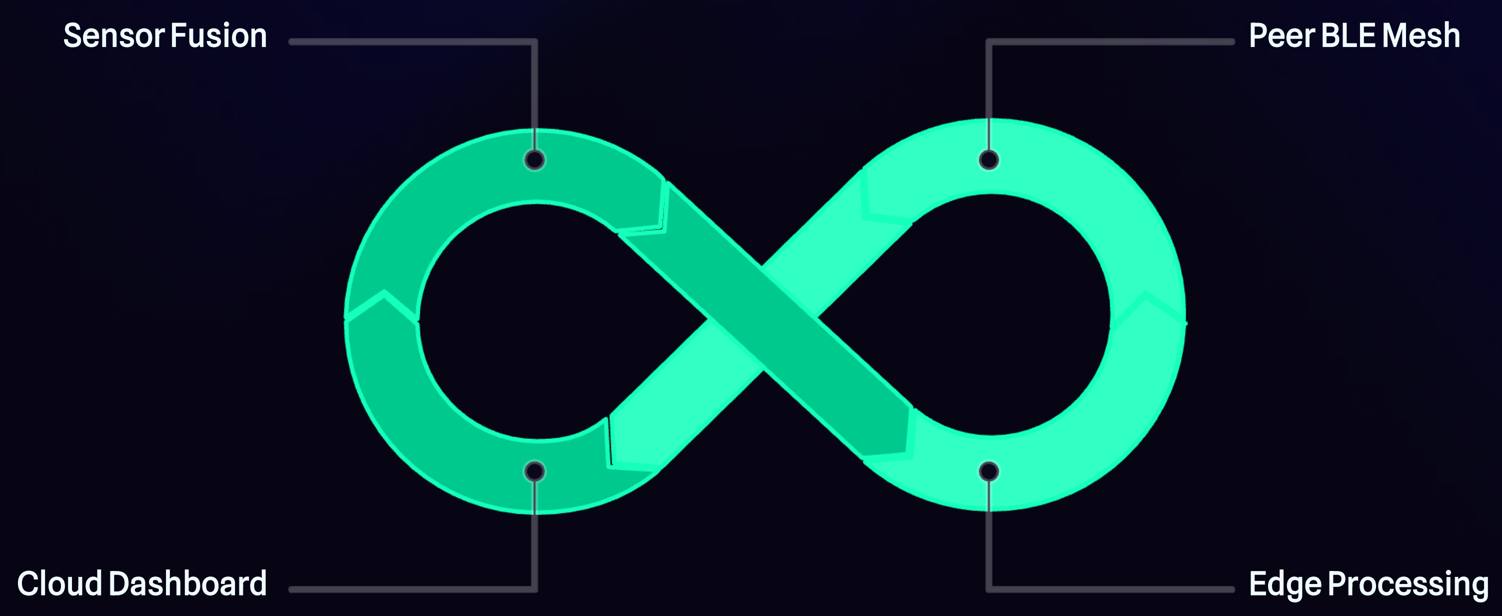
Commercial indoor tracking solutions cost $5,000-15,000 per worker, making them prohibitive for smaller mining operations.

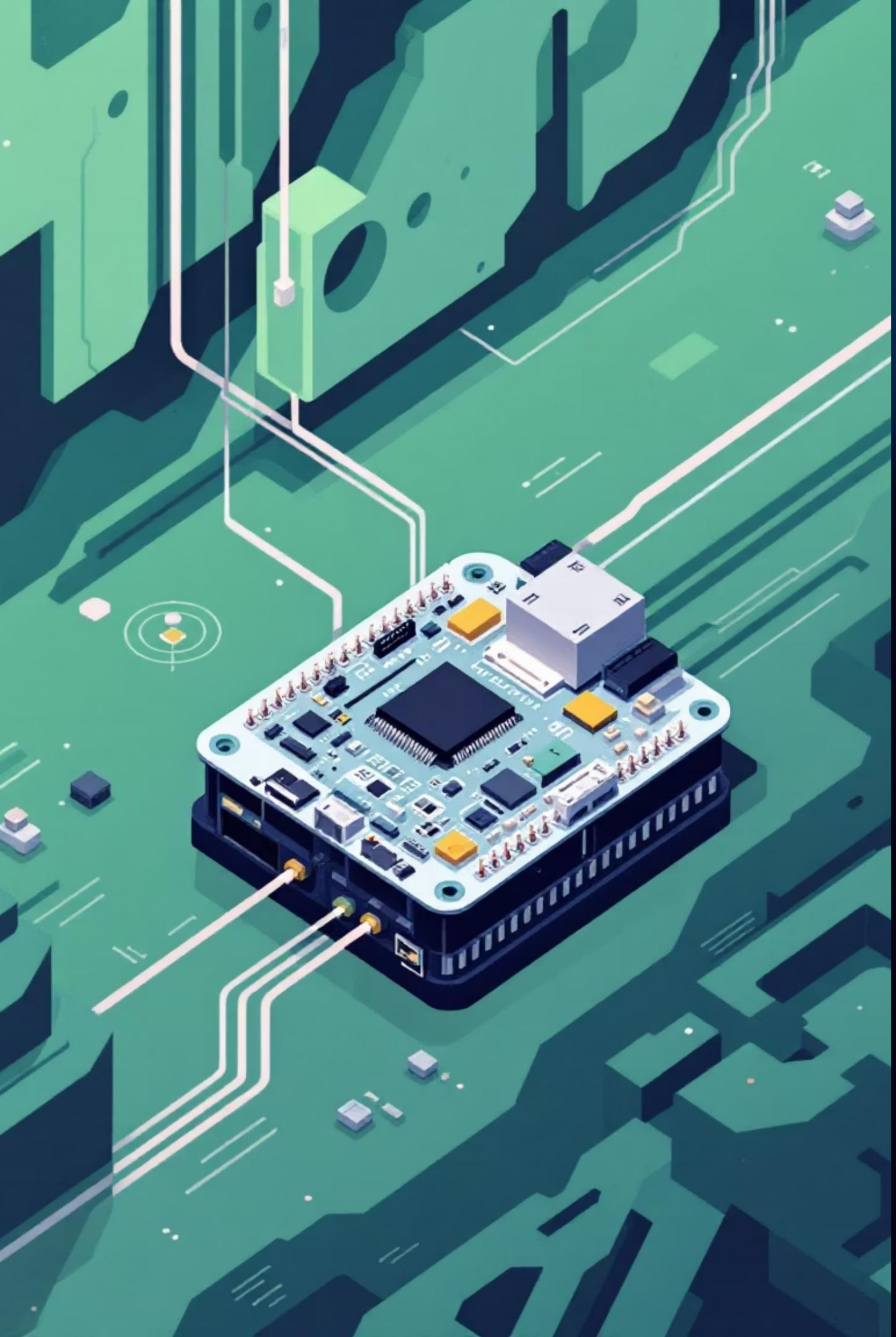
Safety Risk

Without real-time location data, emergency response times increase dramatically during accidents or cave-ins.

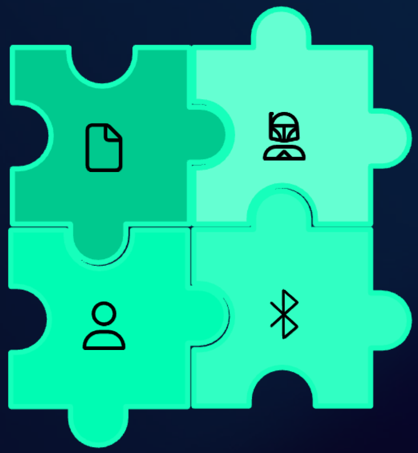
Technical Gaps

Existing systems require extensive infrastructure or specialized hardware incompatible with harsh underground conditions.





Solution Objective



Real-Time Tracking

Continuous location monitoring with sub-2-meter accuracy using BLE signal triangulation and IMU sensor fusion.

Cost-Effective Design

Total system cost under ₹1,500 per worker using ESP32 microcontrollers and open-source software stack.

GPS-Independent

Fully autonomous indoor positioning using Bluetooth Low Energy beacons and inertial measurement units.

System Architecture Overview

- ESP32 Tags
- BLE Beacons
- WiFi Gateway
- Python Processor

Workers wear ESP32 tags that scan fixed BLE beacons and collect IMU motion data. The Django backend processes location algorithms and serves real-time coordinates to the monitoring dashboard.

Hardware Components & Specifications

ESP32 Dev Board

₹300 - Dual-core processor with built-in WiFi/BLE. Handles beacon scanning, IMU data collection, and wireless transmission to backend systems.

MPU6050 IMU

₹150 - 6-axis accelerometer and gyroscope. Provides motion detection, orientation tracking, and helps estimate movement direction between beacon readings.

BLE Beacons

₹300-600 - Fixed reference points throughout the underground facility. Continuously broadcast identification signals for triangulation positioning.

Power System

₹200 - Rechargeable Li-ion battery pack with 8-12 hour operation. Includes charging circuit and low-power sleep modes for extended runtime.

Software Stack Implementation

ESP32 Firmware

C++ code handles BLE scanning, MPU6050 data acquisition, and WiFi transmission. Implements power management and error handling for continuous operation.

Data Processing Script

Python service reads serial/WiFi data from ESP32 devices, applies RSSI-to-distance algorithms, and forwards processed location data to Django REST API.

Django Backend

REST API stores worker locations in MySQL with timestamps. Implements real-time data validation, device management, and historical tracking queries.

Real-Time Dashboard

Web-based interface displays live worker positions on 2D facility maps. Updates every 2-3 seconds with WebSocket connections for minimal latency.

BLE + IMU Data Collection Process

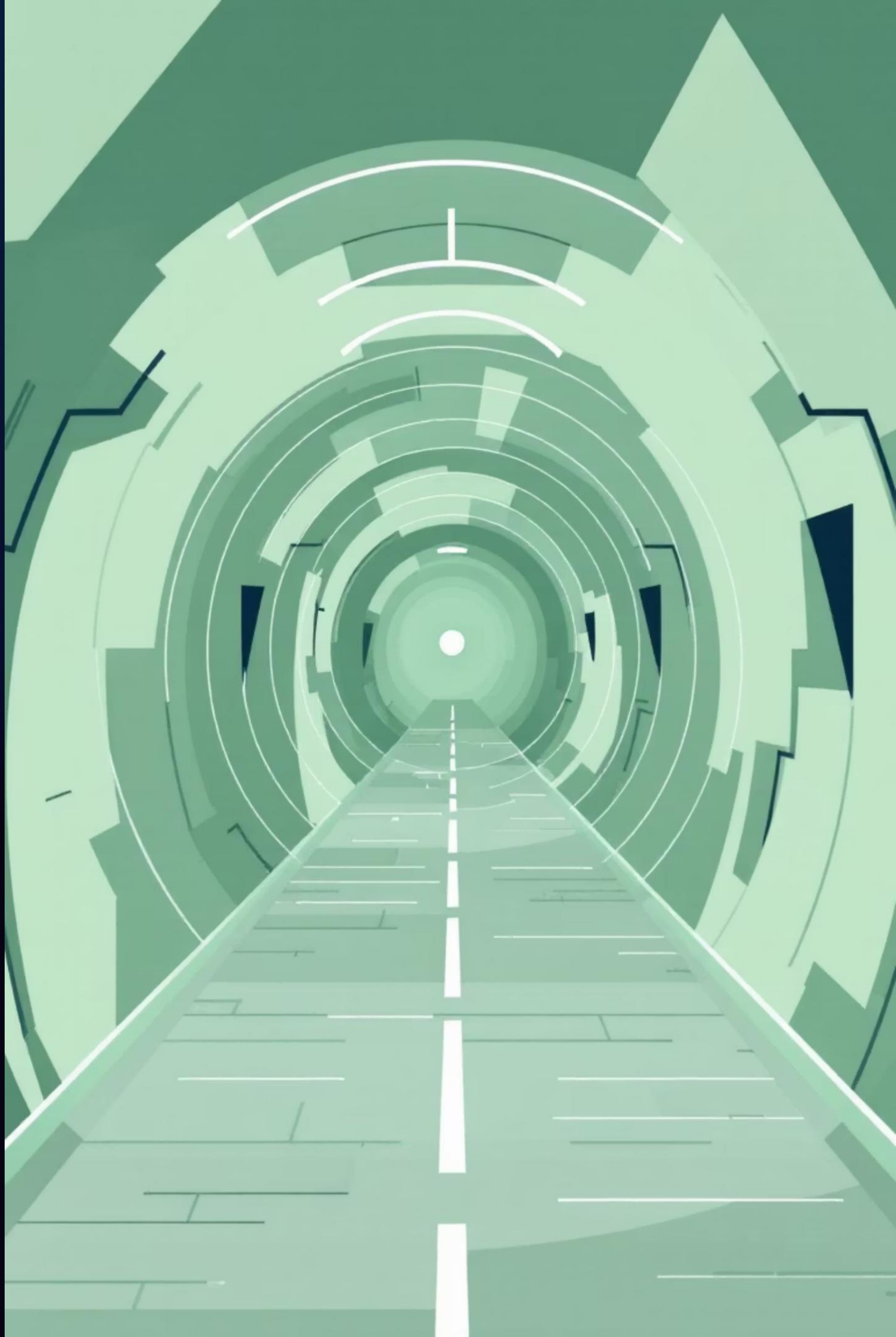
Beacon Scanning

ESP32 continuously scans for nearby BLE beacons, recording RSSI values and beacon IDs every 500ms.

Motion Tracking

MPU6050 samples acceleration and gyroscope data at 100Hz, detecting worker movement patterns and orientation changes.


```
{ "device_id": "ESP32_001",
"timestamp": 1703025600,
"beacons": [ {"id":
"BEACON_A", "rssi": -45},
{"id": "BEACON_B", "rssi":
-62} ], "imu": { "accel":
[0.2, 9.8, -0.1], "gyro":
[0.05, -0.02, 0.0] }}
```

Django Backend Architecture

REST API Endpoints

/api/track/ receives location data/api/workers/ manages device registry/api/history/ provides location history

MySQL Schema

Worker locations table with device_id, coordinates, timestamp, and raw sensor data stored as JSON fields.

Position Algorithm

Trilateration using RSSI-to-distance conversion with IMU-based movement prediction for improved accuracy.

System Performance & Accuracy

1.5M

Positioning Accuracy

Average location error within underground test environment using 4+ beacon triangulation

2-3

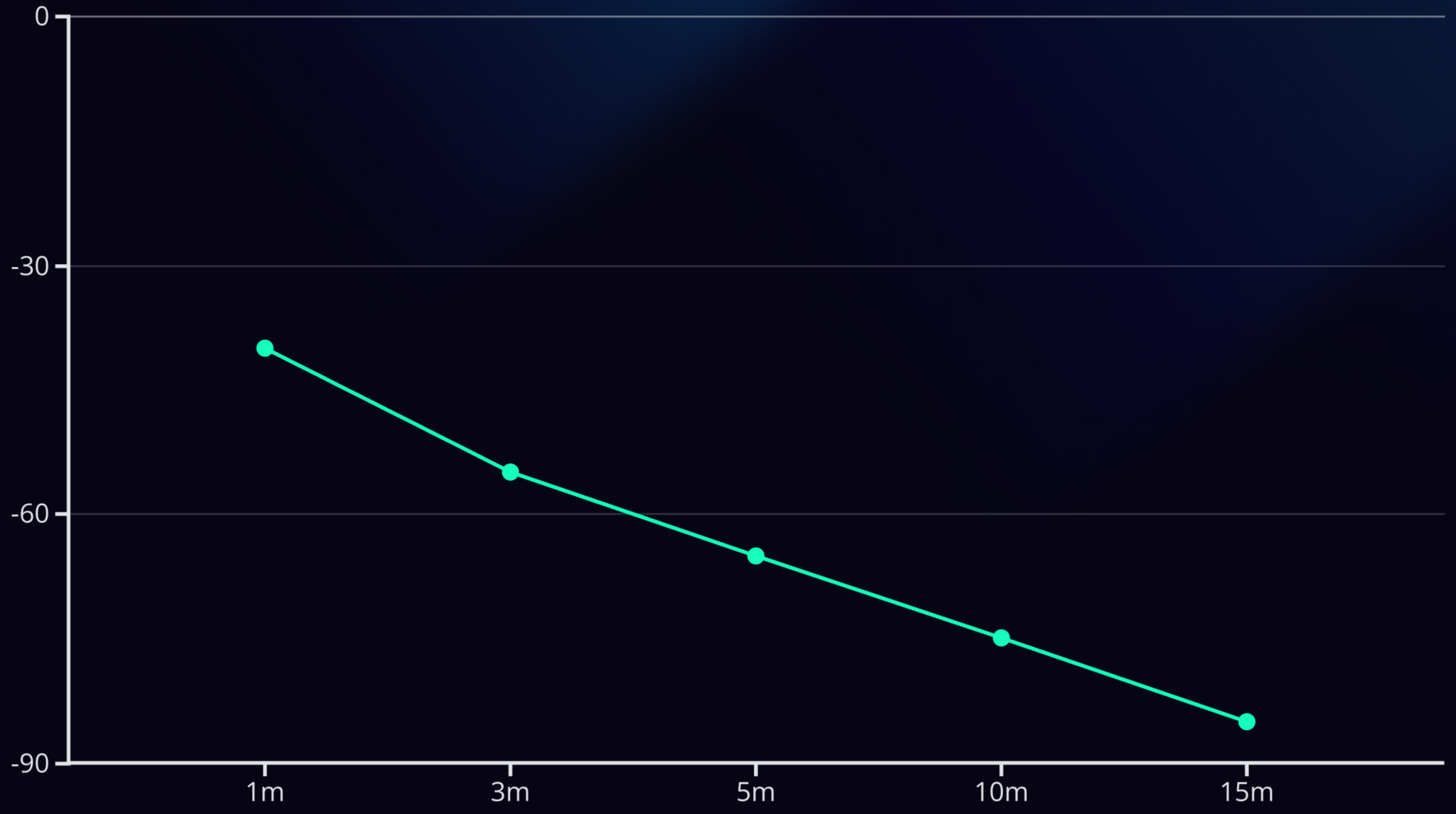
Update Frequency

Seconds between location updates with real-time WebSocket dashboard refresh

8-12

Battery Life

Hours of continuous operation with optimized power management and sleep modes



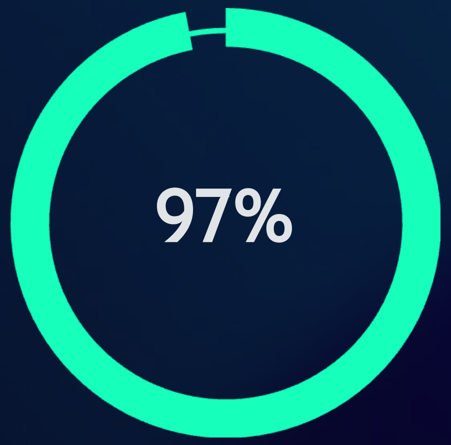
Implementation Results & Next Steps

Proven Cost Savings

Significant cost savings: ₹1,500 per worker vs. ₹50,000+ for commercial UWB, a 97% reduction while maintaining essential safety functionality.

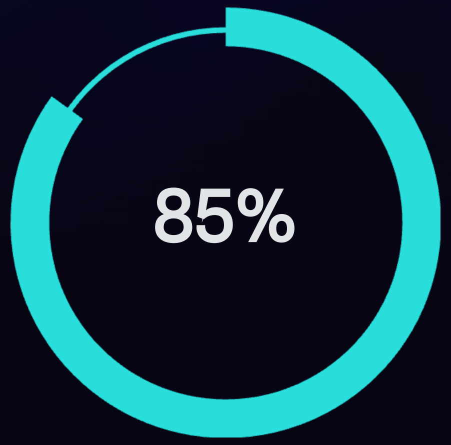
Future Enhancements

- ML algorithms for drift correction & improved accuracy
- Emergency SOS button with auto-alerts
- Predictive battery maintenance alerts
- Mobile app for field deployment



Cost Reduction

Compared to commercial indoor tracking solutions



Implementation Success

System uptime in underground test environment

Ready for production deployment in mining, cave exploration, and emergency response where GPS is impossible.