**Road Infrastructure Management System (RIMS)**

**Professional User Manual**

**Version:** 1.0  
**Date:** April 27, 2025  
**Prepared for:** Mr Ahmed

**Introduction**

Welcome to the Road Infrastructure Management System (RIMS) for Kuwait's Ministry of Public Works. This comprehensive system is designed to transform road infrastructure management through a centralized digital platform that integrates GIS mapping, real-time monitoring, automated maintenance alerts, and multi-stakeholder collaboration.

This manual provides detailed instructions for installing, configuring, and using RIMS to manage road infrastructure projects, maintenance activities, and public reporting.

**System Overview**

RIMS is a comprehensive platform that addresses several key challenges in road infrastructure management:

**Project Goals**

**Centralized Road Infrastructure Management**

* A single digital platform to manage road data, maintenance, and progress tracking
* Easy access to historical and real-time data for informed decision-making

**Improved Maintenance & Road Safety**

* Automated maintenance scheduling and alert systems to prevent deterioration
* Real-time monitoring to improve road safety and reduce risks

**Real-Time Project Tracking & Contractor Oversight**

* Monitor progress, delays, and compliance of infrastructure projects
* Dashboard-based system for reporting, alerts, and issue tracking

**GIS-Based Visualization & Mapping**

* 2D GIS (MDU) mapping for better visualization
* Integrated tools to monitor road conditions, traffic impact, and maintenance schedules

**Efficient Collaboration & Decision-Making**

* Smooth communication among government agencies, contractors, and stakeholders
* Automated analytics and reports for better infrastructure planning

**Key Features**

* **GIS-Based Interactive Maps:** Track projects, maintenance activities, and reported issues
* **Heatmaps & Status Indicators:** Color-coded project progress and risk levels
* **Kanban-Style Task Boards:** Contractor task tracking and management
* **Financial Dashboards:** Budget monitoring and expenditure analytics
* **Public Reporting System:** Community engagement through issue reporting

**Getting Started**

**System Requirements**

* **Hardware:** Standard desktop or laptop computer with internet access
* **Operating System:** Windows 10/11, macOS 10.15+, or Linux
* **Browser:** Chrome 90+, Firefox 88+, Safari 14+, or Edge 90+
* **Internet:** Broadband connection (10+ Mbps recommended)
* **Mobile:** iOS 14+ or Android 10+ for the mobile app

**User Roles and Access Levels**

RIMS supports four primary user roles, each with specific access permissions:

1. **Administrator (Ministry/System Owner)**
   * Full system access and configuration
   * User management and role assignment
   * Project oversight and approval authority
2. **Contractor**
   * Access to assigned projects only
   * Progress reporting and documentation
   * Budget and resource tracking for assigned tasks
3. **Government Agency/Stakeholder**
   * Read access to all projects
   * Approval/rejection authority for specific requests
   * Analytics and reporting capabilities
4. **Public User**
   * Issue reporting access
   * Public project timeline viewing
   * Road condition and closure information

**Installation Guide**

This section provides detailed instructions for installing and configuring RIMS on Windows, macOS, and Linux systems.

**Prerequisites**

Before installing RIMS, ensure you have the following:

* Node.js (version 16.x or higher)
* npm (usually comes with Node.js installation)
* MongoDB (local installation or MongoDB Atlas account)
* VS Code or any code editor of your choice

**Step 1: Extract the ZIP File**

1. Right-click the ZIP file and select "Extract All..." (Windows) or use your preferred extraction tool
2. Choose a destination folder for the extracted files
3. Complete the extraction process

**Step 2: Open the Project in Your Code Editor**

For Visual Studio Code:

1. Open VS Code
2. Select File > Open Folder...
3. Navigate to the extracted project folder
4. Click "Open"

For Other Editors:

* Follow your editor's process for opening a project folder

**Step 3: Install Dependencies**

Open a terminal in your editor (in VS Code: Terminal > New Terminal) and run:

npm install

This will install all dependencies listed in the package.json file. The process might take a few minutes.

**Step 4: MongoDB Setup**

The project is configured to use a MongoDB Atlas connection. You have two options:

**Option 1: Use the existing MongoDB Atlas connection**

* No action required - the connection string is already in the code:
* mongodb+srv://areebanaz4848:pakistanpak12@cluster0.u9vg0.mongodb.net/?retryWrites=true&w=majority&appName=Cluster0

**Option 2: Use your own MongoDB (local or Atlas)**

1. Create a .env file in the project root
2. Add the following line with your connection string:
3. MONGODB\_URI=your\_mongodb\_connection\_string

**Step 5: Starting the Application**

**For Windows Users:**

Windows requires special handling for environment variables. You have several options:

**Option 1: Install cross-env (Recommended)**

1. Install cross-env:
2. npm install --save-dev cross-env
3. Modify the package.json file. Find the "scripts" section and change:
4. "dev": "NODE\_ENV=development tsx server/index.ts"

to:

"dev": "cross-env NODE\_ENV=development tsx server/index.ts"

1. Start the application:
2. npm run dev

**Option 2: Use PowerShell syntax**

$env:NODE\_ENV="development"; npm run dev

**Option 3: Create a bat file** Create a file named start-dev.bat with the following content:

set NODE\_ENV=development

npm run dev

Then run it:

.\start-dev.bat

**Option 4: Fix Windows-specific server binding issues** If you encounter ENOTSUP or similar address binding errors:

1. Open server/index.ts
2. Find the server.listen section (around line 70-80)
3. Change host: "0.0.0.0" to host: "127.0.0.1"
4. Remove the reusePort: true option
5. Save and restart

**For macOS/Linux Users:**

Simply run:

npm run dev

**Step 6: Accessing the Application**

Once the application is running successfully, you'll see console output indicating:

Connected to MongoDB

Database already seeded

MongoDB connected and seeded successfully

[express] serving on port 5000

Open your web browser and navigate to:

http://localhost:5000

**Step 7: Login Credentials**

You can use these pre-configured accounts:

* **Admin:** admin / password
* **Contractor:** contractor / password
* **Ministry:** ministry / password
* **Regular User:** user / password

Or you can register a new account through the registration form.