Janeeb
Solutions

INSTALLATION MANUAL

Extended Planning Instrument for
Unpredictable Spaces and Environments

Introduction

This logistics optimization system is designed to improve the efficiency of goods transportation by using advanced algorithms and machine learning. It optimizes the space inside trucks, reducing costs, minimizing environmental impact, and ensuring that every load is packed with maximum efficiency. This manual will guide you through the installation process.

System Composition

The system is divided into several key components:

- **User Interface:** For logistics managers to input constraints and monitor progress.
- **Dynamic Packing Algorithm:** Adjusts the placement of goods in real-time.
- **Machine Learning Model:** Continuously learns and improves the packing process.
- **Real-Time Dashboard:** Displays real-time data on packing and algorithm performance.
- **Simulation Renderer:** Provides a visual representation of the packing process.

Prerequisites (for development and deployment)

Specific PC Hardware:

- **CPU:** 6 Core CPU or higher (e.g., Intel i5 4690 / AMD Ryzen 5 5600)
- **Memory:** 8 GB or more
- **Storage:** 20 GB free space

Software, Packages & Services:

- **Browser:** Latest version of Google Chrome or Firefox
- **Docker:** Latest version
- **NodeJS:** Version 16 or higher
- **Node Package Manager (NPM):** Version 16 or higher
- **Vue.js:** Version 3
- **PostgreSQL:** Version 12
- **Supabase:** API key setup
- **GitHub Actions:** For CI/CD

General Installation

Step 1: Clone the Repository

To begin the installation process, you'll need to clone the project repository. Open your terminal and run the following command:

git clone https://github.com/COS301-SE-2024/Extended-Planning-Instrument-for-Unpredictable-Spaces-and-Environments.git

Step 2: Navigate to the Project Directory

After cloning the repository, navigate into the project folder by running:

cd Extended-Planning-Instrument-for-Unpredictable-Spaces-and-Environments

Step 3: Install Dependencies

Once inside the project directory, you need to install the necessary dependencies. Run the following command:

npm install

This will download and install all the packages required for the project.

Step 4: Run the Development Server

After installing the dependencies, start the development server with:

npm run dev

This command will compile and run the project. You can then access the application in your browser, typically at `http://localhost:3000` (depending on the configuration).

Install Docker

Docker is crucial for containerizing your application and managing dependencies across environments.

Windows:

- Download Docker Desktop from the Docker website: https://www.docker.com/products/docker-desktop.

- Run the installer and follow the instructions.
- Verify the installation by running `docker --version` in your terminal.

Linux (Debian/Ubuntu):

- Update your package index: `sudo apt-get update.`
- Install Docker: `sudo apt-get install docker-ce docker-ce-cli containerd.io.`
- Start Docker and enable it on boot: `sudo systemctl start docker` and `sudo systemctl enable docker.`
- Verify installation with `docker --version.`

MacOS:

- Download Docker Desktop for Mac from the Docker website: https://www.docker.com/products/docker-desktop.
- Install and verify by running `docker --version` in the terminal.

Install Vue

Vue.js is the front-end framework used in this project.

1. **Install Node.js and npm:**
   - Download from the Node.js website: https://nodejs.org/, as npm is included.
2. **Install Vue CLI globally:**
   - Run: `npm install -g @vue/cli.`
   - Verify with `vue --version.`

Install Supabase

Supabase provides backend services like authentication and database management.

1. **Sign Up and Create a Project:**
   - Visit https://supabase.io/ and create an account.
   - Set up a new project, noting the URL and API keys.
2. **Optional: Install Supabase CLI:**
   - Run: `npm install -g supabase.`
   - Verify with `supabase --version.`
3. **Integrate Supabase into Your Project:**
   - Install the Supabase JavaScript client: `npm install @supabase/supabase-js.`
   - Configure the client using your project URL and API key.

Deployment & Running

To build an run the project enter:

npm start

To access the product service open your browser of choice and navigate to:

http://localhost:3000/

To run all unit tests enter:

npm run test

To run specific test classes enter:

npm run test <class name>
eg
npm run test Packer.test.js

To run tests with code coverage enter:

npm run test:coverage

To start supabse enter:

cd supabase/functions
supabase start

To serve supabase functions enter:

cd supabase/functions
supabase start
supabase functions serve

To deploy supabase functions enter:

cd supabase/functions
supabase start
supabase functions deploy core