

Enobase

Project Overview

Enobase is a Next.js-based web application that provides a powerful formula calculation engine with OpenAI integration. The application offers an interactive canvas-based interface where users can create, manipulate, and visualize data through both manual and AI-assisted workflows.

Features

Interactive Canvas Environment

- Drag-and-drop component workspace
- Add and arrange multiple component types:
 - Variable components for single values
 - Formula components for calculations
 - Data grid components for tabular data
- Real-time updates and calculations

Data Management

- Import data from CSV files into data grids
- Manual data entry and editing
- Variable assignment and management
- Real-time data validation

Formula Capabilities

- Comprehensive formula calculation features:
 - Mathematical operations (add, subtract, multiply, divide)
 - List operations (max, min, mean, median)
 - String manipulations (upper, lower, format)
 - Date operations (format, add, subtract)
- Real-time formula calculation
- Formula dependency tracking
- Error handling and validation

AI Assistant Integration

The application features a powerful AI chat sidebar that enables:

1. Data Generation

- Generate realistic test data based on requirements
- Create sample datasets for testing and development

2. Natural Language Formula Building

- Convert natural language descriptions into formulas
- Use existing canvas data in calculations
- Get explanations of formula logic

3. Automated Visualization

- Generate charts and graphs through natural language requests

- Customize visualizations based on data context

Visualization

- Data visualization using Chart.js
- Multiple chart types support
- Real-time chart updates
- Custom styling options

Modern UI/UX

- Clean, intuitive interface using shadcn/ui
- Responsive design
- Drag-and-drop functionality
- Real-time updates and feedback

Technical Stack

- Frontend: Next.js 15.1.7 with React 19
- UI Framework: Tailwind CSS
- State Management: Zustand
- Charts: Chart.js with react-chartjs-2
- Drag and Drop: dnd-kit
- Type Safety: TypeScript

Prerequisites

- Node.js (Latest LTS version recommended)
- npm or yarn package manager
- OpenAI API key

Setup Instructions

1. Clone the repository:

```
git clone [https://github.com/vibrahim09/enobase_osu.git]
cd formula-engine
```

2. Install dependencies:

```
npm install
```

3. Environment Setup: Create a `.env.local` file in the root directory with the following variables:

```
OPENAI_API_KEY=your_openai_api_key
```

Running the Application

1. Start the Next.js development server:

```
npm run dev
```

The application will be available at `http://localhost:3000`

Build and Deployment

1. Build the application:

```
npm run build
```

2. Start the production server:

```
npm run start
```

Project Structure

```
formula-engine/
├─ src/
│  ├─ app/           # Next.js app directory
│  ├─ components/    # Reusable React components
│  ├─ lib/           # Utility functions and helpers
│  ├─ hooks/         # Custom React hooks
│  ├─ types/         # TypeScript type definitions
│  └─ styles/        # CSS and styling files
├─ public/           # Static assets
├─ docs/             # Documentation
│  ├─ API.md         # API documentation
│  └─ TECHNICAL.md  # Technical documentation
└─ [Configuration files]
```

API Documentation

The Formula Engine provides a RESTful API for various operations. Here are some key endpoints:

Mathematical Operations

```
POST /api/calculate
{
  "function": "multiply",
  "num1": 2,
  "num2": 5
}
```

List Operations

```
POST /api/calculate
{
  "function": "max",
  "numbers": [1, 10, 7]
}
```

String Operations

```
POST /api/calculate
{
  "function": "upper",
  "str": "hello world"
}
```

Date Operations

```
POST /api/calculate
{
  "function": "formatDate",
  "date": "2024-01-01",
  "format": "yyyy-MM-dd"
}
```

For complete API documentation, see `docs/API.md`.

Maintenance Guidelines

Regular Maintenance Tasks

1. Dependency updates
2. Log rotation
3. Performance monitoring
4. OpenAI API quota monitoring

Updating Dependencies

```
npm update
```

Troubleshooting

Common Issues and Solutions

1. Setup Issues

- Run `npm install` to ensure all dependencies are installed
- Verify OpenAI API key is correctly set in `.env.local`
- Check Node.js version compatibility

2. Build Errors

- Clear `.next` directory and `node_modules`
- Run fresh install of dependencies
- Check for TypeScript errors: `npm run type-check`

3. OpenAI Integration Issues

- Verify API key permissions and quota
- Check rate limiting
- Monitor API usage

For detailed troubleshooting guide, see `docs/TECHNICAL.md`.