Digital-Verse

Digital-Verse is an interactive and feature-rich digital circuit simulator designed to make learning and experimenting with digital circuits simple and intuitive.

Tools and Technologies Used

- **Frontend:** React Flow (https://reactflow.dev/), React, Tailwind CSS
- **Backend:** Node.js, Express.js
- **Database:** MongoDB
- **Authentication:** bcrypt, JWT

Features

1. Components Library

Digital-Verse offers a wide range of components to design combinational circuits:

- **Basic Components:** Input/Output nodes, AND, OR, NOT, XOR, NAND, NOR, XNOR gates.
- **Advanced Components:**
  - **7-Segment Display**
  - **Combinational Logic:** Adders, Multiplexers, Decoders.
  - **Sequential Logic:** Clock, Flip-Flops (*Coming soon...*)

2. Boolean Tools

- **Truth Table Generator:** Enter a Boolean expression and get the corresponding truth table, showcasing all possible input combinations and their resulting outputs.
- **Expression Generator:** Provide a truth table and get the simplified Boolean expression. Used the Quine-McCluskey algorithm for minimization.

3. User Account Management

- Register, login, and manage your circuits online.
- Save circuits to your personal account.
- Update passwords or delete accounts securely.

4. Live Editing and Simulation

- Real-time updates to circuit simulation as you make changes.
- Dynamic component interaction.

5. Export Options

- Export circuit in PNG format.

6. Keyboard Shortcuts

- `Shift + Drag`: Select multiple nodes and edges.
- `R`: Rotate selected nodes.
- `Delete`: Delete selected nodes and edges.
- `Ctrl + C`: Copy selected nodes and edges.
- `Ctrl + V`: Paste copied nodes and edges.
- `Ctrl + Alt + R`: Clear the circuit.
- `Ctrl + Alt + K`: Enable/Disable background grid.
- `Ctrl + Alt + J`: Enable/Disable snapping to grid.
- `Ctrl + Alt + L`: Mode selection (light/dark).
- `Ctrl + Shift + E`: Export circuit.
- `Ctrl + Shift + S`: Save circuit.

7. Interaction Guide

- **Drag and Drop Nodes:**
  Simply drag components from the palette and drop them onto the canvas to add them to your circuit. You can easily position and organize them as needed.
- **Making Connections:**
  To connect components, click and drag from the node handles (small circles on the sides of nodes) to another node handle. This creates an edge connecting the two nodes, allowing signals to flow between them.
- **Input Node Value:**
  Click on an **Input** node to change its value. You can toggle between `0` and `1` to simulate different input conditions.
- **Clock Node Frequency:**
  Double-click on a **Clock** node to change its frequency. This allows you to control the timing of sequential circuits, such as flip-flops and other clock-driven components.

Run Locally

Prerequisites

- Node.js installed on your machine.
- A MongoDB database setup (cloud or local).

Installation

1. Clone the repository:

```
git clone https://github.com/ramdhankumar1425/Digital-Verse.git
```

2. Navigate to the project directory:

```
cd Digital-Verse
```

3. Install dependencies:

```
cd Frontend
npm install
cd ..
cd Backend
npm install
```

4. Set up environment variables (`.env` file):

```
// Frontend
VITE_BACKEND_URL  = http://localhost:8000

// Backend
SERVER_PORT  = 8000
CLIENT_URI  = http://localhost:5173
DB_URI = your-database-url
JWT_SECRET = something-secret
```

5. Run the app

```
// Frontend
npm run dev

// Backend
npm run dev
```

Acknowledgments

A huge thank you to:

- Circuit Verse (https://circuitverse.org/) for inspiring this project and setting a foundation for digital circuit simulations.
- The React Flow (https://reactflow.dev/) team for their outstanding library, enabling the seamless and intuitive interface of Digital-Verse.

Contributing

We welcome contributions!

Feel free to submit issues, suggest features, or create pull requests to help improve Digital-Verse.