

Transaction Processing Application

Table of Contents

1. Technology stack and setup Environment
2. Running Frontend and Backend
3. Explanation of Project

1. Technology Stack and Setup Environment

This application uses React for the frontend and Node.js (with Express) for the backend:

1. React: A JavaScript library used for building user interfaces. It helps create a dynamic and responsive frontend for displaying and processing transaction data.
2. Node.js & Express: A JavaScript runtime environment and framework used for handling backend logic. This includes file uploads, parsing CSV files, and calculating transaction balances.

Prerequisites

1. Node.js and npm installed on your machine. You can download them from Node.js (<https://nodejs.org/>).
2. React installed for the frontend.
3. Express.js and related dependencies for the backend.

Backend Setup

1. Clone the repository or download the backend source code.
2. Navigate to the backend directory:
3. Install dependencies for the backend using the following command (in terminal):

```
npm install
```

4. The backend uses multer for file uploads, csv-parser for parsing CSV files, and cors for cross-origin resource sharing. These dependencies are installed with the command above.
5. Make sure port 3000 is available for the backend. You can configure it in the server.js file.

Frontend Setup

1. Navigate to the frontend directory:
2. Install dependencies for the frontend using the following command (in terminal):

```
npm install
```

3. Ensure the frontend is configured to interact with the backend server running on <http://localhost:3000>.

2. Running Frontend and Backend

Running Backend

1. Start the backend by using the following command (in terminal):

```
node server.js
```

2. The backend should now be running on <http://localhost:3000>.

Running Frontend

1. Navigate to the frontend directory:

2. Start the frontend development server using the following command (in terminal):

```
npm start
```

3. The frontend should now be running on <http://localhost:3001> (or another port if 3001 is in use).

4. The frontend will automatically connect to the backend running on <http://localhost:3000>.

3. Explanation of Project

The application contains several tabs in the UI, each serving a specific purpose for reviewing and managing transactions:

1. Uploaded CSV Files

Displays a list of all the CSV files that have been uploaded. After uploading a file, the filename will appear here.

2. Balance Report

Shows a detailed list of all accounts and their corresponding card numbers with the cumulative balance.

Each account is listed with all associated cards, and the total balance is shown for each card.

3. Transaction History

Lists all transactions that were processed from the CSV file, showing details like Account Name, Card Number, Transaction Amount, Transaction Type, Description, and Target Card Number (for transfers).

4. Negative Balance

Displays accounts with cards that have a balance less than 0.00. These accounts are flagged for review.

5. Bad Transactions

Lists all transactions that could not be properly parsed or processed due to missing or invalid data.