💰 Expense Tracker App

A modern, responsive expense tracking application built with React and Tailwind CSS. Track your income and expenses with a beautiful, intuitive interface.

Show Image

✨ Features

🏠 Home Dashboard

- Balance overview with gradient card design
- Monthly income/expense summary with percentage changes
- Recent transactions preview
- Quick access to all features

💳 Transaction Management

- Add new transactions (income/expenses)
- Edit existing transactions
- Delete transactions with confirmation
- Real-time balance calculation
- Category-based organization

🔍 Smart Filtering & Search

- Search by transaction name or description
- Filter by category (All, Income, Expenses)
- Sort by date (newest first)
- Real-time results

📊 Financial Analytics

- Category-wise spending breakdown
- Monthly income/expense tracking
- Transaction statistics
- Visual progress indicators

💾 Data Persistence

- Automatic save to browser localStorage
- Data persists between sessions
- Export functionality ready
- Demo data included

🚀 Getting Started

Prerequisites

- Node.js (v14 or higher)
- npm or yarn

Installation

1. **Clone or create the project:**

   ```bash
   npx create-react-app expense-tracker
   cd expense-tracker
   ```

2. **Install dependencies:**

   ```bash
   npm install lucide-react
   # or
   yarn add lucide-react
   ```

3. **Replace the src folder with provided files**

4. **Start development server:**

   ```bash
   npm start
   # or
   yarn start
   ```

5. **Open your browser:** Navigate to `http://localhost:3000`

📁 Project Structure

```
expense-tracker/
├── public/
│   ├── index.html        # HTML template
│   └── manifest.json     # PWA manifest
├── src/
│   ├── components/
│   │   ├── common/       # Reusable components
│   │   │   ├── BottomNavigation.js
│   │   │   ├── Header.js
│   │   │   └── Modal.js
│   │   ├── transactions/ # Transaction components
│   │   │   ├── TransactionForm.js
│   │   │   ├── TransactionList.js
│   │   │   └── TransactionCard.js
│   │   ├── views/        # Main view components
│   │   │   ├── HomeView.js
│   │   │   ├── TransactionsView.js
│   │   │   └── StatsView.js
│   │   └── ui/           # UI components
│   │       ├── BalanceCard.js
│   │       ├── MetricCard.js
│   │       └── CategoryIcon.js
│   ├── utils/            # Utility functions
│   │   ├── calculations.js # Financial calculations
│   │   ├── storage.js    # LocalStorage operations
│   │   └── constants.js  # App constants
│   ├── hooks/            # Custom React hooks
│   │   ├── useTransactions.js
│   │   └── useStorage.js
│   ├── styles/           # CSS files
│   │   ├── globals.css
│   │   └── components.css
│   ├── App.js            # Main app component
│   ├── index.js          # React entry point
│   └── index.css         # Base CSS
├── package.json          # Dependencies and scripts
└── README.md             # This file
```

🎯 How to Use

Adding Transactions

1. Click the **"+"** button in the bottom navigation
2. Fill in transaction details:
    - Name (required)

    - Amount (required)
    - Category (required)
    - Date (required)
    - Description (optional)

3. Click **"Add Transaction"**

Managing Transactions

- **Edit**: Click the edit icon on any transaction
- **Delete**: Click the trash icon to remove a transaction
- **Search**: Use the search bar to find specific transactions
- **Filter**: Use category filters to narrow down results

Viewing Statistics

- Navigate to the **Stats** tab
- View total income, expenses, and balance
- See category-wise spending breakdown
- Monitor transaction patterns

🎨 Customization

Adding New Categories

Edit `src/utils/constants.js`:

```javascript
export const CATEGORIES = [
  // Add your custom categories
  { id: 9, name: 'Travel', type: 'expense', icon: '✈️', color: 'bg-indigo-100' },
  { id: 10, name: 'Freelance', type: 'income', icon: '💻', color: 'bg-green-100' },
];
```

Changing Colors

Modify Tailwind classes in components or add custom CSS in `styles/globals.css`.

Adding New Features

- Extend the `useTransactions` hook
- Create new components in the appropriate folders
- Add new views to the main navigation

🛠 Technical Details

Built With

- **React 18** - Frontend framework
- **Tailwind CSS** - Utility-first CSS framework
- **Lucide React** - Beautiful icons
- **LocalStorage API** - Data persistence

Key Features

- **Component-based architecture**
- **Custom React hooks** for state management
- **Responsive design** for mobile and desktop
- **Real-time calculations**
- **Form validation**
- **Error handling**

📱 Mobile Support

The app is fully responsive and includes:

- Touch-friendly interface
- Mobile-optimized navigation
- Responsive layouts
- Swipe gestures ready

🚢 Deployment

Build for Production

```bash
npm run build
# or
yarn build
```

Deploy Options

- **Netlify**: Drag and drop the `build` folder
- **Vercel**: Connect your GitHub repository
- **GitHub Pages**: Use `gh-pages` package

- **Firebase Hosting**: Use Firebase CLI

🤝 Contributing

1. Fork the repository
2. Create a feature branch
3. Make your changes
4. Test thoroughly
5. Submit a pull request

📄 License

This project is open source and available under the MIT License.

🆘 Support

If you encounter any issues:

1. Check the console for errors
2. Ensure all dependencies are installed
3. Clear browser cache/localStorage
4. Create an issue with details

🔮 Future Enhancements

- □ Cloud synchronization
- □ Budget planning
- □ Recurring transactions
- □ Export to CSV/PDF
- □ Multi-currency support
- □ Dark mode toggle
- □ Push notifications
- □ Data backup/restore

Happy tracking! 💰 📈