Split App - Backend Assignment

# 🔧 Project Overview

This project is a backend system for managing group expenses, similar to apps like Splitwise. Users can add expenses, split them by amount/percentage, and track who owes whom.

# 🛠️ Tech Stack

- Node.js + Express  
- PostgreSQL (hosted on Railway)  
- Postman (API testing)  
- Deployed via Render

# ✅ Core Features

- Add, update, delete expenses  
- Automatically manage people involved  
- View current balances  
- Calculate settlements (who owes whom)  
- Input validation and error handling

# 📨 API Endpoints

Base URL: `https://split-apps-backend.onrender.com`

**Expense Management**

- `POST /expenses` – Add new expense

- `GET /expenses` – List all expenses

- `PUT /expenses/:id` – Update expense

- `DELETE /expenses/:id` – Delete expense

**People & Settlements**

- `GET /people` – List all people involved

- `GET /balances` – Net balance for each person

# 📦 Sample Request

POST /expenses  
```  
{  
 "amount": 600,  
 "description": "Dinner",  
 "paid\_by": "Shantanu",  
 "split\_type": "exact",  
 "shares": {  
 "Shantanu": 200,  
 "Sanket": 200,  
 "Om": 200  
 }  
}  
```

# 📬 Postman Collection

- All API endpoints are included in a publicly accessible Postman collection.  
- Collection URL: https://github.com/ranadive-25/splitwise-backend/blob/main/Split%20App%20-%20DevDynamics.postman\_collection.json

# 🚀 Deployment

- The backend is deployed on Render  
- Public base URL is available in the Postman environment and used in all requests

# 📌 Notes

- `amount` and `share` are stored and processed in rupees.  
- Data is validated on server-side.  
- Error messages and status codes are returned for invalid operations.