

Name	Employee Code
------	---------------

# digit SPLITTER

An Expense Management System

Made by- Ankur Pandey

Harshita Mahajan

Sairaj Rajput

Sapna Rani

# ABOUT OUR PROJECT

The Expense Management System is a web application built with Angular that helps users manage shared expenses within groups.

- Helps users manage shared expenses within groups.
- Users can create groups and add members.
- Users can add expenses and view all the groups they are part of.
- The system automatically calculates how much each user owes, ensuring transparency in expense sharing.
- Authentication is integrated using JWT tokens for secure access.
- Supports both admin and user roles.
- Exceptions are managed using an HTTP interceptor for reliable error handling.



# ARCHITECTURE OVERVIEW

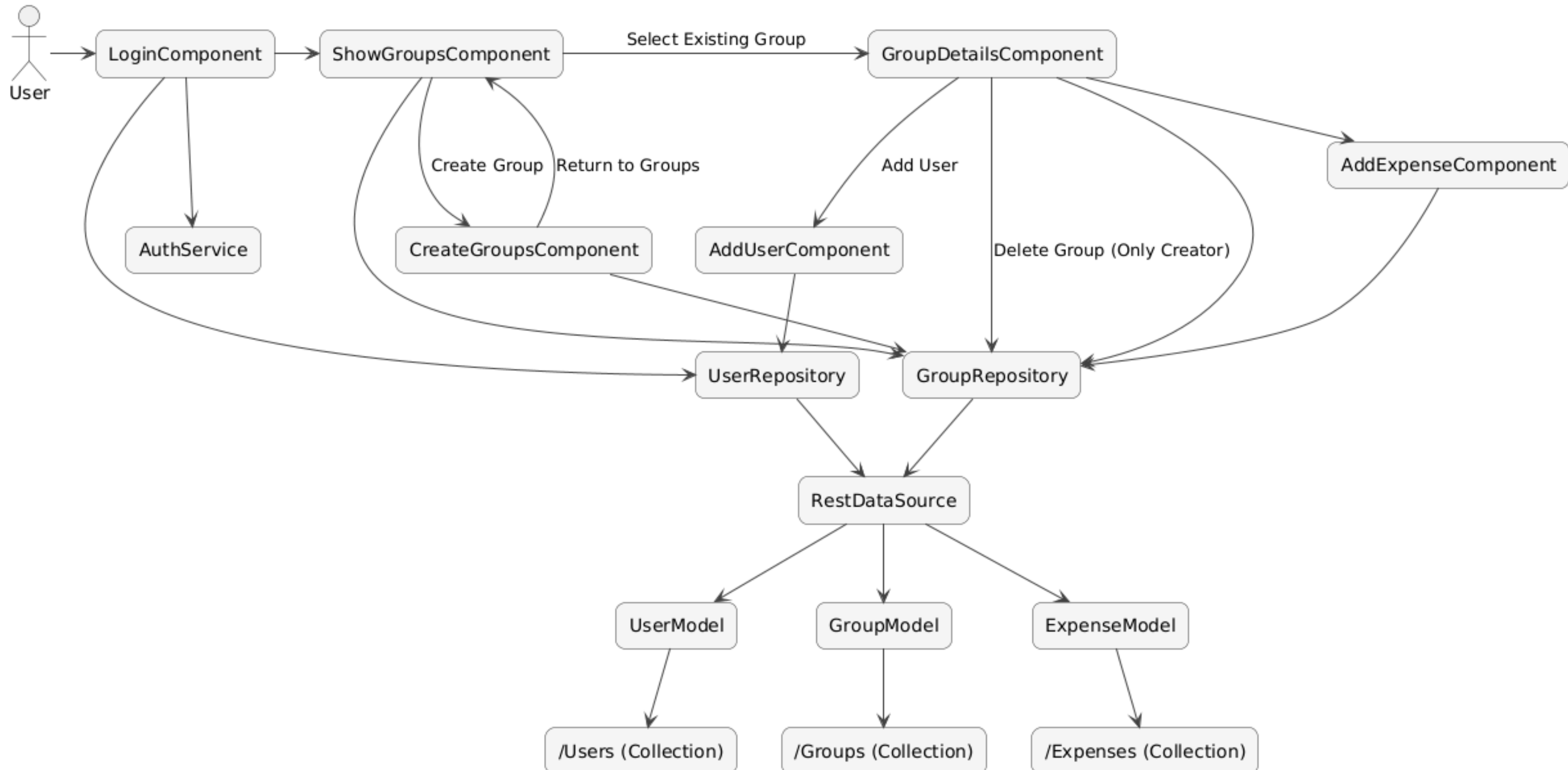
The app is structured in a feature-first style with clear separation of concerns:

- Auth → JWT-based authentication, middleware for protecting routes.
- Backend → server.js with Express API, authMiddleware.js for security and serverData.json for mock persistence.
- Group → Create groups and list all groups the user is part of.
- Expense → Add expenses within groups, calculate balances (owed/owes).
- Core → HTTP interceptor for exception handling, guards for authentication.
- Shared → Common Angular components, services, and utilities reused across modules.
- Build/Deploy → Production build artifacts inside dist/expense-tracking-project.
- PWA → Configured with ngsw-config.json for service worker support.

Routing: Routes cover authentication, group listing, and expense logging. Guards ensure authentication before accessing feature modules. Backend middleware enforces token-based security on API endpoints.

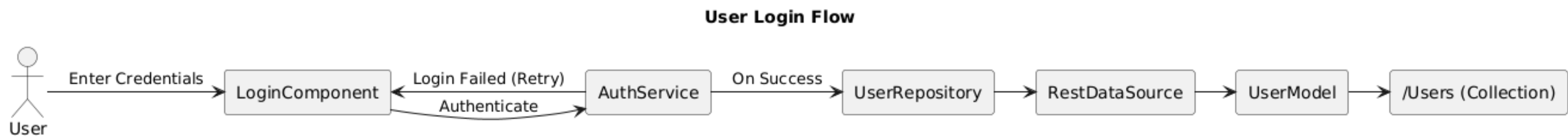
# FLOWCHART

Application Process Flow

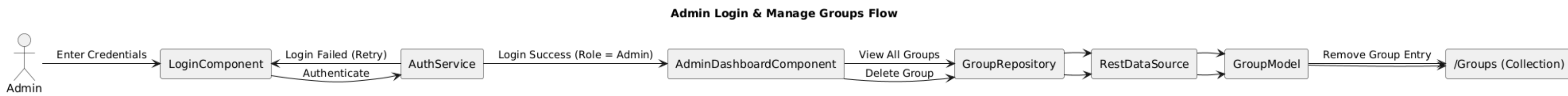


# FLOWCHART

## USER LOGIN



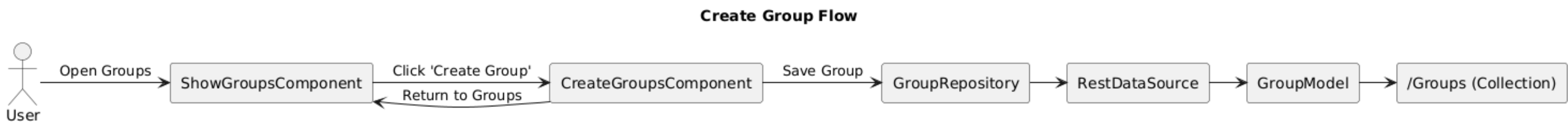
## ADMIN LOGIN



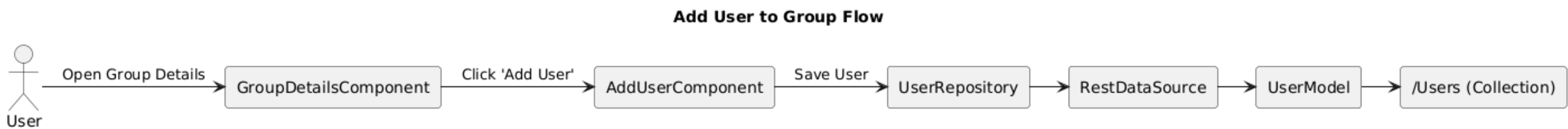


# FLOWCHART

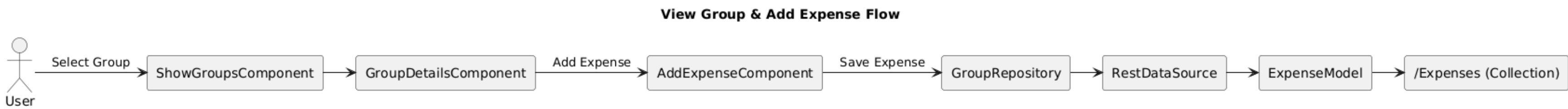
## CREATE GROUP



## ADD USER



## ADD EXPENSE



**THANK YOU**

---