■ 3-Week Roadmap: Personal Expense Tracker

Week 1 – Foundation: Auth + Backend + DB

Goal: Have a working backend with auth & DB connected.

- Project Setup: Create GitHub repo, init Node.js project with Express/Fastify.
- Authentication (AWS Cognito): Create User Pool, enable signup/signin, write /auth/check route to validate JWT.
- Database (Postgres local): Design schema (users, categories, expenses), connect with pg/Prisma.
- APIs: POST/GET categories, POST/GET expenses, middleware to enforce JWT validation.

■ *End of Week 1: Postman-tested APIs with Cognito auth + Postgres persistence.*

Week 2 – Frontend: React + Integration

Goal: Have a basic working UI for tracking expenses.

- Frontend Setup: Init React/Next.js app, add Cognito login/signup forms.
- Dashboard: Protected route, add expense form, list expenses in a table.
- Filters: Filter expenses by category/date.

■ *End of Week 2: You can log in, add expenses, and see them in a dashboard.*

Week 3 – Deployment + Extras

Goal: Deploy a live version + polish.

- Deployment: Backend → AWS Lambda + API Gateway (or EC2), Database → AWS RDS (Postgres), Frontend → S3 + CloudFront.
- Stretch (Optional): Add admin role (Cognito claim), Admin-only reports API, Expense summary chart, CI/CD with GitHub Actions.

■ *End of Week 3: Live demo URL + GitHub repo → fully portfolio-ready.*

■ Stretch Options (if you finish early)

- Add CSV export for expenses.
- Add tags for expenses.
- Add AWS CloudWatch monitoring for APIs.