# 🎯 Project Requirements Document (PRD)

## Project Name:

GiftGenius – AI-Powered Gift Recommendation System

## 1. Project Overview

GiftGenius is a RAG-based (Retrieval-Augmented Generation) gift recommendation system designed to help users find personalized gifts based on the recipient's interests, occasion, and budget. It leverages AI and smart database structuring to make accurate and fast gift suggestions via a responsive web application.

## 2. Goals & Objectives

🎁 Provide Smart Gift Recommendations using RAG-based AI.
⚡ Quick and Efficient Searches based on user filters like interests, age, budget, and occasion.
🌐 Deploy a Fully Functional Web App with backend API integration and a scalable database.
📦 Store Gift Data and User Search History in a hosted MySQL database.

## 3. Features

### Core Features

☑ AI-powered gift suggestions based on user input.
☑ Search filters for budget, occasion, gender, and age group.
☑ Display detailed gift information with images, descriptions, ratings, and purchase links.
☑ Maintain user search history for better recommendations over time.

### Technical Features

☑ RAG-based model for AI gift suggestion.
☑ Node.js + Express.js backend with REST APIs.
☑ Hosted MySQL database on Railway for 24x7 accessibility.
☑ Responsive frontend with HTML, CSS, and JavaScript.
☑ GitHub Pages hosting for frontend deployment.

## 4. Technical Stack

| Component | Technology |
|---|---|
| Frontend | HTML, CSS, JavaScript |
| Backend | Node.js, Express.js |
| AI/ML | RAG-based Recommendation Engine |
| Database | MySQL (Hosted on Railway) |
| Deployment | GitHub Pages, Railway |
| Collaboration | GitHub |

## 5. System Architecture

### Components:

- Frontend (Static Website)
- Backend API (Node.js + Express)
- MySQL Database (Hosted on Railway)
- RAG-based AI Model

### Data Flow:

1. User submits gift search request.
2. Frontend sends API request to backend.
3. Backend queries MySQL database & AI model.
4. Backend responds with gift recommendations.
5. Frontend displays results to user.

## 6. Database Schema

Tables:

- users – Stores user info.
- categories – Gift categories.
- gifts – Gift details including price, image, and purchase links.
- gift_requests – Stores user search input.
- user_search_history – Logs gift suggestions provided to users.

## 7. API Endpoints

| Method | Endpoint | Description |
|---|---|---|
| GET | `/` | Health check |
| GET | `/products` | Fetch all gifts |
| GET | `/products_paginated` | Fetch gifts with pagination |

| Method | Endpoint | Description |
|---|---|---|
| POST | `/recommendations` | Get gift recommendations (RAG-based) |

## 8. Roles & Responsibilities

| Person | Role | Tasks |
|---|---|---|
| Frontend Dev | UI/UX & Website Integration | Build and connect frontend to backend |
| Backend Dev | API Development | Develop Node.js APIs & database queries |
| AI/ML Dev | AI Model & Recommendation | Build and train RAG-based suggestion engine |
| Database & Deployment | Database Management & Hosting | Set up Railway database and deploy the backend |

## 9. Deployment Plan

Frontend: Hosted on GitHub Pages
Backend: Hosted on Railway (Node.js app)
Database: Hosted on Railway (MySQL)
Version Control: GitHub repository with full team access

## 10. Security Considerations

- Do not hardcode database credentials. Use Railway environment variables.
- Regularly check for credential leaks using tools like GitGuardian.
- Ensure CORS is properly configured to only allow frontend origin.

## 11. Timeline

| Phase | Task | Owner | Status |
|---|---|---|---|
| Phase 1 | Database design & setup (local) | Database person | ✅ Completed |
| Phase 2 | Populate initial data | Database person | ✅ Completed |
| Phase 3 | Backend API setup (local testing) | Backend Developer | ✅ Completed |
| Phase 4 | Database hosting on Railway | Database person | ✅ Completed |
| Phase 5 | Backend deployment on Railway | Backend Developer | ✅ Completed |
| Phase 6 | Frontend development & API connection | Frontend Developer | In Progress |

| Phase | Task | Owner | Status |
|---|---|---|---|
| Phase 7 | Final Integration & Testing | Entire Team | Pending |