Excelente iniciativa! Criar uma aplicação de lista de tarefas com Laravel é um ótimo projeto para consolidar conhecimentos em **autenticação, relacionamento entre modelos, controle de permissões, CRUD e boas práticas de organização de código**.

Aqui vai um **plano passo a passo**, com **ideias para aprimorar** o projeto — **sem mostrar nenhum código**, como pediste:

**✅ 1. Planejamento do Projeto**

**Objetivo geral:**  
Cada usuário pode se registrar, fazer login e gerenciar suas próprias tarefas (criar, editar, concluir ou apagar), além de poder editar seus dados pessoais.

**Funcionalidades principais:**

* Registro e login de usuários
* Criação, visualização, edição e exclusão de tarefas
* Marcar tarefas como concluídas
* Edição de dados do perfil do usuário
* Sistema que impede acesso às tarefas de outros usuários

**Funcionalidades extras que podem melhorar o projeto:**

* Upload de avatar no perfil
* Filtros de tarefas por “concluída” ou “pendente”
* Sistema de prioridade nas tarefas
* Notificações por email quando uma tarefa estiver para vencer
* Interface mobile responsiva (pode usar Bootstrap ou Tailwind)
* Modo escuro/claro
* Paginação das tarefas

**🛠️ 2. Estrutura Inicial**

**Etapas sugeridas:**

1. **Criar um novo projeto Laravel**
   * Usa o laravel new ou composer create-project.
2. **Configurar o banco de dados**
   * Define o nome do banco no .env.
3. **Gerar autenticação**
   * Usa o sistema de autenticação pronto do Laravel com Breeze, Jetstream ou Laravel UI.
   * Isso já te dá rotas, views e lógica para login e registro.

**🧩 3. Modelagem dos Dados**

**Tabelas básicas que vais precisar:**

* users: onde ficam os dados dos usuários (nome, email, senha, etc.)
* tasks: onde cada tarefa está associada a um usuário

**Relacionamento:**

* Um usuário pode ter muitas tarefas
* Cada tarefa pertence a um único usuário

**🗂️ 4. Funcionalidades CRUD**

Cria páginas e formulários para:

* Ver todas as tarefas do usuário atual
* Adicionar nova tarefa
* Editar uma tarefa
* Marcar como concluída (pode ser um botão)
* Deletar uma tarefa

**Extras recomendados:**

* Adicionar campo "data de vencimento" na tarefa
* Campo “prioridade” (baixa, média, alta)
* Exibir tarefas em ordem cronológica ou por prioridade

**🔐 5. Proteção e Acesso**

* Garante que **cada usuário só veja suas tarefas**.
* Usa **middleware de autenticação** para proteger as rotas.
* Usa o **ID do usuário logado** para filtrar as tarefas que aparecem.
* Bloqueia acesso manual via URL às tarefas de outros usuários.

**👤 6. Perfil do Usuário**

* Cria uma área para o usuário editar:
  + Nome
  + Email
  + Senha
  + Avatar (opcional)

**Dicas:**

* Pode permitir upload de imagem de perfil
* Mostra a foto e nome no cabeçalho da aplicação

**📈 7. Interface e Experiência do Usuário**

* Usa Bootstrap ou Tailwind para um visual limpo
* Adiciona ícones com FontAwesome ou Heroicons
* Cria mensagens de feedback (ex: “Tarefa criada com sucesso”)
* Responsividade para dispositivos móveis

**🧪 8. Testes e Melhorias**

* Testa login e registro
* Tenta acessar tarefas de outros usuários e garante que não é possível
* Verifica se tarefas são salvas corretamente e se a edição funciona
* Melhora a performance com cache (opcional)

**🚀 9. Implantação**

Quando o projeto estiver pronto:

* Usa serviços como **Laravel Forge**, **Render**, **Railway** ou **Vercel** (com Docker)
* Configura as variáveis de ambiente e o banco de dados na nuvem

Se quiser, depois posso te dar o **roteiro completo com os comandos e arquivos** que precisas criar. E, quando fores implementar uma parte específica, posso te explicar os **códigos detalhados**, passo a passo.

Queres começar por qual etapa? Posso te acompanhar do início ao fim.