This task outlines the process of setting up a Laravel project, creating backend APIs for users and projects, and providing guidance on branching, committing, and creating a pull request (PR) using Git. Here's a step-by-step breakdown:

1. **Setup Laravel Project: (Done)**
   * Download the provided initial code from the Google Drive link.
   * Extract the contents of the downloaded directory into a new directory named **task-13-laravel** within your project structure.
2. **Switch to Main Branch and Add Code:(Done)**
   * Navigate to the main(master) branch.
   * Add the code from the extracted directory to the main branch.
   * Commit the changes with the message "Starting Laravel Project".
3. **Create a New Branch:(Done)**
   * Create a new branch from the main(master) branch as prescribed by the Git Snippets.
   * This new branch will be where you perform your work.
4. **Develop Backend APIs:**
   * Implement the following endpoints:
     + **/users**: Lists all users in the database.
     + **/user/:userId/projects**: Lists all projects for the specified user.
   * Optionally, you can create additional endpoints for tasks like creating a new user or project, or deleting a user or project.
5. **Seeder for Dummy Data:**
   * Create a seeder to generate dummy data for both users and projects.
   * Seeders are scripts that populate the database with sample data.
6. **Documentation and Resources:**
   * Refer to Laravel documentation (version 9.x) and PHP manual for guidance on Laravel APIs and PHP language features.
7. **Optional:**
   * Implement additional endpoints beyond the required ones, based on project requirements or preferences.
8. **Running Docker Commands:**
   * Utilize the provided guide for running Docker commands directly if needed.
9. **Acceptance Criteria:**
   * Ensure that the developed app allows listing all users and viewing projects of a specific user.
10. **Finishing Up:**
    * Commit your code to the branch you created for this task.
    * Create a pull request (PR) with your changes for review.

Following these steps should lead to the successful completion of the task, with a functioning Laravel backend providing the required APIs for users and projects. Make sure to adhere to coding standards, documentation guidelines, and best practices throughout the development process.