

Create a Project called `ToDoList` using laravel framework. The ToDoList Web Page need to meet the requirement below:

1. Create an **Authentication** for the user to login.
2. Users are able to **create, retrieve, update, and delete** tasks.
3. Use **Roboto** as Primary Font Style. (<https://fonts.google.com/specimen/Roboto>)
4. You need to **publish the project file to GitHub** Repository

Additional:

1. You can used any package that help the development
2. You free to design the web page according to your creativity
3. Best if you can used Vue Js Framework for front end development (**Optional**)

Below are guides that can help in order to perform the task but **you are free to use any technique required.**

1. Install Laravel
 - a. Set Up Laravel Project
 - b. Create a new Laravel project named `ToDoList`.
 - c. Install Laravel Breeze Starter Kit
2. Authentication using Laravel Breeze – Blade/Vue
3. Change Font and Style
 - a. Modify Fonts
 - b. Customise your application's CSS in `resources/css/app.css`. For instance, change the background colour of the body and modify button styles
4. Create a To-Do List
 - a. Create Model and Migration named `Task`
 - i. Generate a model and migration for `Task`.
 - ii. Define the schema for the tasks table in the migration file. Example below:

```
Schema::create('tasks', function (Blueprint $table) {  
    $table->id();  
    $table->string('title');  
    $table->boolean('is_completed')->default(false);  
    $table->foreignId('user_id')->constrained()->onDelete('cascade');  
    $table->timestamps();  
});
```

- b. Create Controller and Routes for `Task`
 - i. Generate a controller for handling tasks
 - ii. Define routes in routes/web.php example as below.

```
Route::middleware(['auth'])->group(function () {  
    Route::get('/tasks', [TaskController::class, 'index'])->name('tasks.index');  
    Route::post('/tasks', [TaskController::class, 'store'])->name('tasks.store');  
    Route::put('/tasks/{task}', [TaskController::class, 'update'])->name('tasks.update');  
    Route::delete('/tasks/{task}', [TaskController::class, 'destroy'])->name('tasks.destroy');  
});
```

- c. Implement Task Controller Methods
 - d. Create Task Views where users can perform insert, update and delete actions.
5. Submission Instructions
- a. Repository Setup
 - i. Create a public repository on GitHub
 - ii. Push the code to the repository
 - b. ReadMe File
 - i. Include a README.md file with instructions on how to set up and run the project locally
 - c. Submission
 - i. Share the link to the GitHub repository.