1. app/Http/Controllers/BookingController.php

**Need Approvement:**

To be honest when I start grow through the code, I am confusing its booking system or application portal. In a code it look a Job Portal where user can buy pro or a guest access on job applications. User receiving emails and notifying either user already applied or not and what will be status of jobs. User can find categories of job professional, paid and unpaid jobs in on portal. And each user can select translator to ready JDS. User can apply different jobs in code basically defined jobs process. Application can reopen and also calculate distance between jobs.

**Email template:**

We must Use Email Boostrape template in it will be easy and look professionally reference (**acceptJobWithId). Function name be not be conversional.**

**Translator:**

you can use Google translator as well in it to translate language because in coding we must use international standards.

$response['status'] = 'fail';

$response['message'] = 'Du har redan en bokning den tiden ' . $job->due . '. Du har inte fått denna tolkning';

**Hard Coded:**

Status must be Enum in database not like pending, accepting rejecting. It should be 0,1,2 etc. We must use date properly dynamically by using php function.

if (isset($requestdata['to']) && $requestdata['to'] != "") {

$to = $requestdata["to"] . " 23:59:00";

$allJobs->where('due', '<=', $to);

}

**Code comments:**

Code must have comments so that it will be helpful for next developer to study code.

**Controller Model :**

Each category or each curd must have own controller and methods if we use same classes and files it is not a way to use MVC Frame work.

**Alerts:**

We can use socket for alerts it is basically an event listener that’s help us to provide alerts to users.

**Admin/ Super Admin:**

In code super admin and admin must have separated files so that if we are alerting a code then only one side of admin stop and other admin can work.

if ($cuser && ($cuser->is('superadmin') || $cuser->is('admin'))) {

$allJobs = DB::table('jobs')

}