PROJECT #2: “Laravel Self-Directed Project”

In groups of one or two, create a Laravel project implementing the skills you’ve learned throughout the course.

The subject of the project is up to you - you may choose to create a personal portfolio, a blog website or take on a more ambitious project; but regardless, the scope must be first approved by the professor.

You will have three weeks to complete the project - its final due date being Monday, April 15th, 2024.

|  |  |
| --- | --- |
| Week 1 | Monday March 25 – Sunday March 31 |
| Week 2 | Monday April 1 – Sunday April 7 |
| Week 3 | Monday April 8 – Sunday April 14 |

This project is worth 10% of your final grade. The project will be partly assessed based upon the inclusion of the requirements provided below. Note: regardless of the scope, all projects must meet the minimum requirements stated below. All development must be performed on the school’s php server (scweb.ca).

All remaining classes are dedicated to group work on the project. All group members are expected to attend each class to work on the project. You may also require additional time outside of class to complete this project.

During each class, you will be asked to demonstrate your project and update the professor as to its completion status. You may also wish to use this time to ask questions about issues that you have encountered.

This project is your opportunity to show off your Laravel skills. Laravel is a very powerful and feature-rich framework so don’t be afraid to explore the full breadth and depth that it offers.

Instructions

Begin by selecting a partner; groups of two are recommended. If you would like to work on your own, you must have prior approval from the professor.

You and your groupmate must answer the following questions and submit the responses through the submission point on blackboard before the due date.

1. Group member information:

|  |  |
| --- | --- |
| Student Name | Student Number |
| Ex. Nicholas Sylvestre | 01234567 |
| Ex. Jane Doe | 76543210 |

1. Briefly explain the project scope including duties assigned to each group member:

Example: We plan to develop a blog similar to the project from third semester.

Administrators of the website will be able to login and post new blog posts and create/retrieve/update/delete new categories of posts.

Administrators will be able to upload custom avatars which will be displayed within the title of the blog post.

Standard users will be able to add comments to a blog post but cannot create new blog posts.

We will use the bootstrap framework for from end design.

Nicholas will work primarily with the front-end design including all views, master pages and view partials. Jane Doe will be responsible for routes, models. Both Nicholas and Jane will be responsible for application logic throughout the project

Minimum Requirements

Routes

* Project incorporates user authentication in a meaningful way
* All routes forward to appropriate controllers
* All routes are defined using Resource Routes

Controllers

* All application logic is performed in appropriate controllers
* Project contains CRUD actions for each resource
* Project uses RESTful best practices with regards to Routes and Resource Controller actions

Views

* All views use Blade directives (No <?php tags)
* Project uses Blade Master Templates
* Project Form Views use the following helper methods: action(), old(), csrf\_field(), asset()
* All forms must perform appropriate validation
* Forms must incorporate Form Request Classes to perform validation
* Forms must include logic to output errors returned from an invalid input
* Project incorporates View Partials where appropriate
* Project uses View Composers of the View Façade to share data with all views

Models

* Project incorporates the use of a database
* Project uses models and the Eloquent ORM to perform database actions
* Project uses Laravel recommendations with regards to Model names
* Project uses Route-Model Binding where appropriate
* Project incorporates either One-to-Many and One-to-Many Inverse or Many-to-Many a relationship
* Project incorporates Guards/Auth middleware to prevent unauthorized access to resources

Database

* Project uses migrations to perform database DDL actions (create/update tables)
* Project uses seeding to generate test data.
* Project relationships incorporate referential integrity through foreign key restraints

If any of requirements are unclear, please contact the professor for further explanation