Creating an ASP.NET MVC e-commerce website to sell laptops involves several steps. Below is a step-by-step process to create this project without providing the actual code:

Step 1: Project Setup

- Open Visual Studio and create a new ASP.NET MVC web project.
- Choose the appropriate .NET framework version.
- Select the Empty template and check the "MVC" and "Web API" checkboxes.
- Click on "Create" to generate the project.

Step 2: Database Setup

- 2.1. Install SQL Server 22 Express Edition or a later version if not already installed.
- 2.2. Create a new database for your e-commerce website.
- 2.3. Define the tables needed for your application. At a minimum, you will need tables for products, customers, orders, and order details.
- 2.4. Establish a connection to your SQL Server database from your MVC project using Entity Framework or a Data Access Layer (DAL).

Step 3: Model Creation

- 3.1. Create model classes that represent the tables in your database. For example, create a Product class, a Order class, etc.

Step 4: Views

- 4.1. Use Bootstrap to design your website's user interface. Make sure that all pages are mobile-responsive.
- 4.2. Create views for different sections of your e-commerce site, such as product listing, order details, and user account management.

Step 5: Controllers

- 5.1. Create controllers for different parts of your application, such as a ProductController, OrderController, OrderDetailsController.
- 5.2. Implement actions within each controller to handle user requests, such as displaying product details, adding items to the cart, processing orders, and managing user accounts.

Step 6: Routing

- 6.1. Configure routing in your RouteConfig.cs file to ensure that URLs are mapped to the correct controller actions.

Step 7: Authentication and Authorization

- 7.1. Implement user authentication and authorization to secure sensitive actions and views. You can use ASP.NET Identity for this purpose.

Step 8: Git Integration

- Set up a Git repository for your project on GitHub or another version control platform.
- Commit your code regularly to the repository, and encourage your team to collaborate using Git for version control.

Step 9: Testing and Debugging

- Thoroughly test your application, checking for bugs and ensuring that all features work as expected.

Step 10: Deployment

- Choose a hosting provider for your ASP.NET MVC application, configure your server, and deploy your website to a production environment.

Step 11: Documentation

- Create detailed documentation for your project, including how to set it up, how to use it, and any additional information that may be helpful for future development or maintenance.

Step 12: Continuous Improvement

- Continuously monitor your e-commerce website for performance issues, security vulnerabilities, and customer feedback. Make necessary improvements and updates as needed.

GitHub Repository Link:

https://github.com/rastogi102/P3SQL_PhaseEndProjects.git