

## Image Upload Web Application (Using ASP.NET)

### Home Controller File: Manage request related to Home page

- 3 Namespaces:  
**ImageUploadApp.Models** (import models from ImageUploadApp)  
**Microsoft.AspNetCore.Mvc**, (Import the MVC framework which allows controller logic to manage certain requests)  
**System.Diagnostics** (Used to get diagnostics reports related to error pages)
- Defines the controller name and will act as Model-View-Controller (MVC) that handles incoming HTTP requests and returns appropriate HTML pages.
- **Logger** allowing the app to log information like errors or diagnostics info
- **Public IActionResult Index()**: Handles the request to the homepage and returns the default Index view.
- **Public IActionResult Privacy()**: Handles the request to the privacy page and returns the privacy view.
- **Error()**: This handles the error page.

### Image Upload Controller File: Handling the Upload, Display, and Download part of images.

- Imports various models and input/output operations
- Define a controller to handle the image upload feature of the web app
- Specify the allowed file extensions (JPEG, JPG, PNG, DICOM)
- Establish the path for upload directories and if it doesn't exist then create (wwwroot/upload)
- **HttpGet**: this action is invoked when a user visits the upload image page.
- When this action invokes it constructs a list of ImageModel objects and each model contains; ImageName, ImagePath, ImageType
- Each model is passed as a List and it allows view the images in a grid or gallery view
- **HttpPost**: this action handles the request where the user can upload one or multiple files.
- It checks that the file is nonnull and fulfills the file length greater than 0 conditions.
- It validates the file type with allowed file types and if file format is unsupported then an error is added to the model and the Index view returns with an error message.
- If file is supported and valid then it saved to the wwwroot/upload path.

**Error View Model:** It is a simple model class that represent information about errors in ASP.NET web applications.

**Image Model:** This file defines a class called ImageModel, which is used to represent the metadata of an image in your image upload web application. This model holds details like the image's name, path, and type, which are likely displayed in the gallery or used when processing uploaded images.

**Home View File: Index.cshtml:** This file is a Razor View for the home page of your ASP.NET Core web application. It provides a simple user interface where visitors are welcomed to the Image Upload App

**Image Upload View File:** This file is a Razor View for the Image Upload page of your application. It handles the image upload functionality and displays a gallery of the uploaded images. Users can select images to upload, see a gallery of images, and click to view an image in full size. The page uses the ImageModel class to display and manage the image data.

**Program.cs:** This file is the entry point for your ASP.NET Core web application. It sets up the web application, configures essential services like routing and static files, and handles some specific behaviors like clearing uploaded images on startup.

- Configurations
- Delete uploaded file when startup the applications
- Configures Middleware: Redirects HTTP to HTTPS, server static files, handling routes and reaches the correct request on the click