

# ABCRetailers Web Application - UploadController Code Snapshots

## UploadController.cs (Full Code Snapshot)

```
UploadController.cs X
ABCRetailers ABCRetailers.Controllers.UploadController

1 // Controllers/UploadController.cs
2 using Microsoft.AspNetCore.Mvc;
3 using ABCRetailers.Models;
4 using ABCRetailers.Services;
5
6 namespace ABCRetailers.Controllers
7 {
8     1 reference
9     public class UploadController : Controller
10     {
11         private readonly IAzureStorageService _storageService; // [X] Added underscore
12
13         0 references
14         public UploadController(IAzureStorageService storageService)
15         {
16             _storageService = storageService; // [X] Added underscore
17         }
18
19         0 references
20         public IActionResult Index()
21         {
22             return View(new FileUploadModel());
23         }
24
25         [HttpPost]
26         [ValidateAntiForgeryToken]
27         0 references
28         public async Task<IActionResult> Index(FileUploadModel model)
29         {
30             if (ModelState.IsValid)
31             {
32                 try
33                 {
34                     if (model.ProofOfPayment != null && model.ProofOfPayment.Length > 0)
35                     {
36                         // Upload to blob storage
37                         var fileName = await _storageService.UploadFileAsync(model.ProofOfPayment, "payment-proofs"); // [X] Added underscore
38
39                         // Also upload to file share for contracts
40                         await _storageService.UploadToFileShareAsync(model.ProofOfPayment, "contracts", "payments"); // [X] Added underscore
41
42                         TempData["Success"] = $"File uploaded successfully! File name: {fileName}";
43
44                         // Clear the model for a fresh form
45                         return View(new FileUploadModel());
46                     }
47                     else
48                     {
49                         ModelState.AddModelError("ProofOfPayment", "Please select a file to upload.");
50                     }
51                 }
52                 catch (Exception ex)
53                 {
54                     ModelState.AddModelError("", $"Error uploading file: {ex.Message}");
55                 }
56             }
57
58             return View(model);
59         }
60     }
61 }
```

## UploadController.cs (IAzureStorageService Action Code Snapshot)

```
private readonly IAzureStorageService _storageService; // ☒ Added underscore

0 references
public UploadController(IAzureStorageService storageService)
{
    _storageService = storageService; // ☒ Added underscore
}
```

## UploadController.cs (Index Action Code Snapshot)

```
0 references
public IActionResult Index()
{
    return View(new FileUploadModel());
}

[HttpPost]
[ValidateAntiForgeryToken]
0 references
public async Task<IActionResult> Index(FileUploadModel model)
{
    if (ModelState.IsValid)
    {
        try
        {
            if (model.ProofOfPayment != null && model.ProofOfPayment.Length > 0)
            {
                // Upload to blob storage
                var fileName = await _storageService.UploadFileAsync(model.ProofOfPayment, "payment-proofs"); // ☒ Added underscore

                // Also upload to file share for contracts
                await _storageService.UploadToFileShareAsync(model.ProofOfPayment, "contracts", "payments"); // ☒ Added underscore

                TempData["Success"] = $"File uploaded successfully! File name: {fileName}";

                // Clear the model for a fresh form
                return View(new FileUploadModel());
            }
            else
            {
                ModelState.AddModelError("ProofOfPayment", "Please select a file to upload.");
            }
        }
        catch (Exception ex)
        {
            ModelState.AddModelError("", $"Error uploading file: {ex.Message}");
        }
    }

    return View(model);
}
```