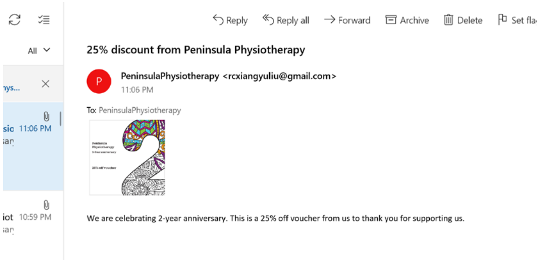
FIT5032 Assignment 3

Xiangyu Liu 27503410

Implementation of file upload and email with attachment.

FIT5032_27503410/PeninsulaPhysiotherapy at main · RoshanLiu/FIT5032_27503410 (github.com)

Bulk Email Example



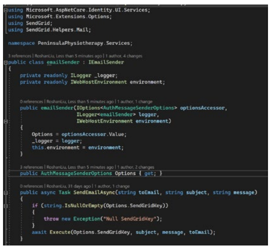
25% discount from Peninsula Physiotherapy

PeninsulaPhysiotherapy <rcxiangyuliu@gmail.com>
11:06 PM

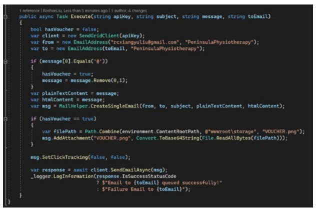
To: PeninsulaPhysiotherapy

We are celebrating 2-year anniversary. This is a 25% off voucher from us to thank you for supporting us.

EmailSender.cs

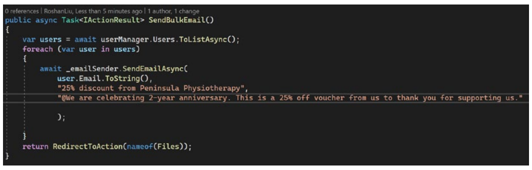


```
using Microsoft.AspNetCore.Identity.UI.Services;
using Microsoft.Extensions.Options;
using SendGrid;
using SendGrid.Helpers.Mail;

namespace PeninsulaPhysiotherapy.Services;

3 references | RoshanLiu, Less than 5 minutes ago | 1 author, 4 changes
public class emailSender : IEmailSender
{
    private readonly ILogger _logger;
    private readonly IWebHostEnvironment environment;


    0 references | RoshanLiu, Less than 5 minutes ago | 1 author, 1 change
    public emailSender(IOptions<AuthMessageSenderOptions> optionsAccessor,
                       ILogger<emailSender> logger,
                       IWebHostEnvironment environment)
    {
        Options = optionsAccessor.Value;
        _logger = logger;
        this.environment = environment;
    }

    3 references | RoshanLiu, Less than 5 minutes ago | 1 author, 2 changes
    public AuthMessageSenderOptions Options { get; }

    0 references | RoshanLiu, 31 days ago | 1 author, 1 change
    public async Task SendEmailAsync(string toEmail, string subject, string message)
    {
        if (string.IsNullOrEmpty(Options.SendGridKey))
        {
            throw new Exception("Null SendGridKey");
        }
        await Execute(Options.SendGridKey, subject, message, toEmail);
    }
```



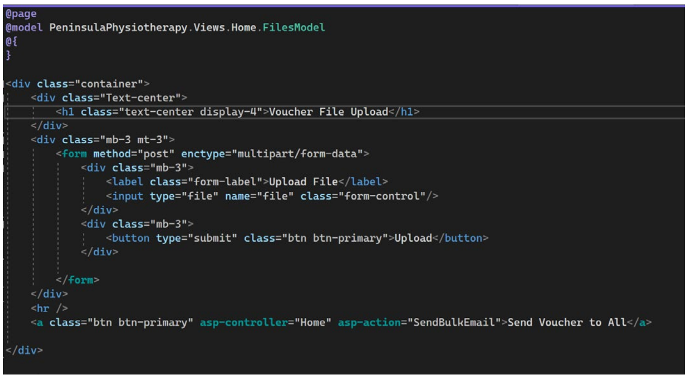
```
    1 reference | RoshanLiu, Less than 5 minutes ago | 1 author, 4 changes
    public async Task Execute(string apiKey, string subject, string message, string toEmail)
    {
        bool hasVoucher = false;
        var client = new SendGridClient(apiKey);
        var from = new EmailAddress("rcxiangyuliu@gmail.com", "PeninsulaPhysiotherapy");
        var to = new EmailAddress(toEmail, "PeninsulaPhysiotherapy");

        if (message[0].Equals('@'))
        {
            hasVoucher = true;
            message = message.Remove(0,1);
        }
        var plainTextContent = message;
        var htmlContent = message;
        var msg = MailHelper.CreateSingleEmail(from, to, subject, plainTextContent, htmlContent);

        if (hasVoucher == true)
        {
            var filePath = Path.Combine(environment.ContentRootPath, @"wwwroot\storage", "VOUCHER.png");
            msg.AddAttachment("VOUCHER.png", Convert.ToBase64String(File.ReadAllBytes(filePath)));
        }

        msg.SetClickTracking(false, false);

        var response = await client.SendEmailAsync(msg);
        _logger.LogInformation(response.IsSuccessStatusCode
                               ? $"Email to {toEmail} queued successfully!"
                               : $"Failure Email to {toEmail}");
    }
}
```

Home Controller (Part)



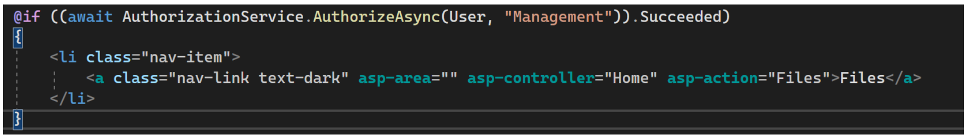
```
0 references | RoshanLiu, Less than 5 minutes ago | 1 author, 1 change
public async Task<IActionResult> SendBulkEmail()
{
    var users = await userManager.Users.ToListAsync();
    foreach (var user in users)
    {
        await _emailSender.SendEmailAsync(
            user.Email.ToString(),
            "25% discount from Peninsula Physiotherapy",
            "@We are celebrating 2-year anniversary. This is a 25% off voucher from us to thank you for supporting us."

            );

    }
    return RedirectToAction(nameof(Files));
}
```

View



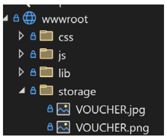
```
@page
@model PeninsulaPhysiotherapy.Views.Home.FilesModel
@{
}

<div class="container">
    <div class="Text-center">
        <h1 class="text-center display-4">Voucher File Upload</h1>
    </div>
    <div class="mb-3 mt-3">
        <form method="post" enctype="multipart/form-data">
            <div class="mb-3">
                <label class="form-label">Upload File</label>
                <input type="file" name="file" class="form-control"/>
            </div>
            <div class="mb-3">
                <button type="submit" class="btn btn-primary">Upload</button>
            </div>

        </form>
    </div>
    <hr />
    <a class="btn btn-primary" asp-controller="Home" asp-action="SendBulkEmail">Send Voucher to All</a>

</div>
```

Shared View



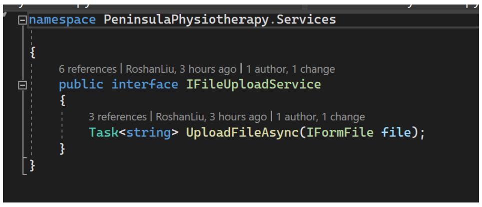
```
@if ((await AuthorizationService.AuthorizeAsync(User, "Management")).Succeeded)
{
    <li class="nav-item">
        <a class="nav-link text-dark" asp-area="" asp-controller="Home" asp-action="Files">Files</a>
    </li>
}
```

Files



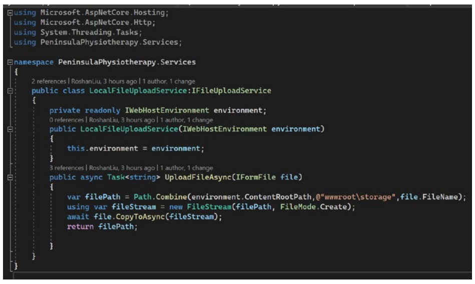
```
wwwroot
    css
    js
    lib
    storage
        VOUCHER.jpg
        VOUCHER.png
```

IFileUploadService.cs



```
namespace PeninsulaPhysiotherapy.Services

{
    6 references | RoshanLiu, 3 hours ago | 1 author, 1 change
    public interface IFileUploadService
    {
        3 references | RoshanLiu, 3 hours ago | 1 author, 1 change
        Task<string> UploadFileAsync(IFormFile file);
    }
}
```

LocalFileUploadService.cs



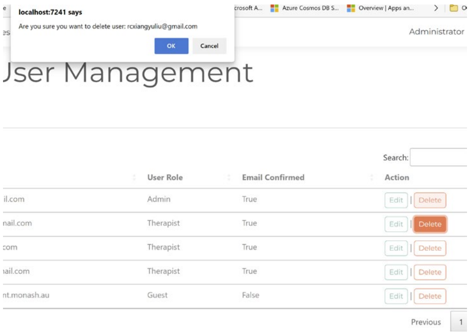
```
using Microsoft.AspNetCore.Hosting;
using Microsoft.AspNetCore.Http;
using System.Threading.Tasks;
using PeninsulaPhysiotherapy.Services;

namespace PeninsulaPhysiotherapy.Services
{
    2 references | RoshanLiu, 3 hours ago | 1 author, 1 change
    public class LocalFileUploadService:IFileUploadService
    {
        private readonly IWebHostEnvironment environment;
        0 references | RoshanLiu, 3 hours ago | 1 author, 1 change
        public LocalFileUploadService(IWebHostEnvironment environment)
        {
            this.environment = environment;
        }
        3 references | RoshanLiu, 3 hours ago | 1 author, 1 change
        public async Task<string> UploadFileAsync(IFormFile file)
        {
            var filePath = Path.Combine(environment.ContentRootPath,@"wwwroot\storage",file.FileName);
            using var fileStream = new FileStream(filePath, FileMode.Create);
            await file.CopyToAsync(fileStream);
            return filePath;

        }
    }
}
```

Implementation of any one innovative feature of your choice aimed at improving User Experience (UX) and inclusion of a user story

User story:

As an administrator, I want to have a change to regret when I accidently clicked the delete button, so that I can save the record that I don't want to delete.

Innovative feature: There is a delete confirm dialog appears when delete button is clicked. The user in story can press cancel to abort deleting.



localhost:7241 says

Are you sure you want to delete user: rcxiangyuliu@gmail.com

Jser Management

Search:

	User Role	Email Confirmed	Action
il.com	Admin	True	Edit | Delete
nail.com	Therapist	True	Edit | Delete
com	Therapist	True	Edit | Delete
ail.com	Therapist	True	Edit | Delete
nt.monash.au	Guest	False	Edit | Delete

[Template for Studio Assessment Task 3]
{To be submitted in Moodle}

1. Implementation of file upload and email with attachment

2. Implementation of any one innovative feature of your choice aimed at improving User Experience (UX) and inclusion of a user story.

Declaration

I declare that this Studio Assessment Task 3 report is my individual work. I have not copied from any other student’s work or from any other source except where due acknowledgment is made explicitly in the report, nor has any part of this submission been written for me by another person.

Signature: Xiangyu liu