Note that the following code is built on top of code from the week 8 lab (Will incorporate this into my repository later on).

Implementation of file upload and email with attachment:

Graphical user interface

Description automatically generated

Figure 1: Email Contents

Graphical user interface, application

Description automatically generated

Figure 2: Email Sent

Graphical user interface, text, email

Description automatically generated

Figure 3: The Corresponding Email:

Controller Endpoint That Recieves the Email Contents & Uploaded File. <https://github.com/3drdsh3in/FIT5032/blob/master/lab8/FIT5032_Week08A/FIT5032_Week08A/Controllers/HomeController.cs>

Email Sender (Adapted to encode the InputStream as Base64 before sending):

<https://github.com/3drdsh3in/FIT5032/blob/master/lab8/FIT5032_Week08A/FIT5032_Week08A/Utils/EmailSender.cs>

Send Email Form:

<https://github.com/3drdsh3in/FIT5032/blob/master/lab8/FIT5032_Week08A/FIT5032_Week08A/Views/Home/Send_Email.cshtml>

Implementation of any one innovative feature aimed at improving User Experience (UX)

What I have done in this task is just a simple POC in react.js & ASP.NET Core Framework that enables client & staff to authenticate & login (no additional features beyond this).

Note: I didn’t have time to finish this.

User Story: “As a staff, I want to be able to login/authenticate into the web application across any platform, so that I do not have to exclusively use my windows desktop when trying to interact with my appointments”.