**Final Project Check-in #1**

**Unit-Testing**

The unit tests for an email management application work just like a safety net. Whenever we modify code in an application, we can run a set of unit tests to check whether the modification breaks existing functionality. Unit tests make the code safe to modify. Unit tests make all of the code in your application more resilient to change. We will create the following two unit tests for the email management system.

1. Creating Unit Tests for the Service Layer

Let s start by creating a set of unit tests for our Email amangemt application s service layer. We'll use these tests to verify our validation logic.

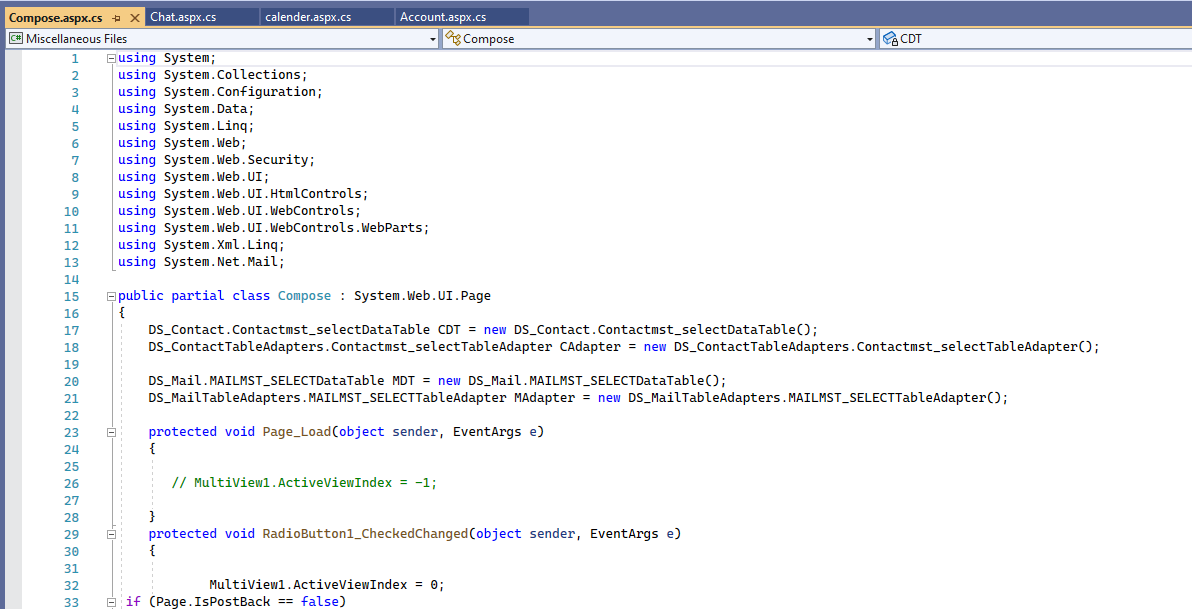
1. Creating Unit Tests for Controllers

Email management system applications which created on c # control the flow of user interaction. When testing a controller, you want to test whether the controller returns the right action result and view data. You also might want to test whether a controller interacts with model classes in the manner expected.

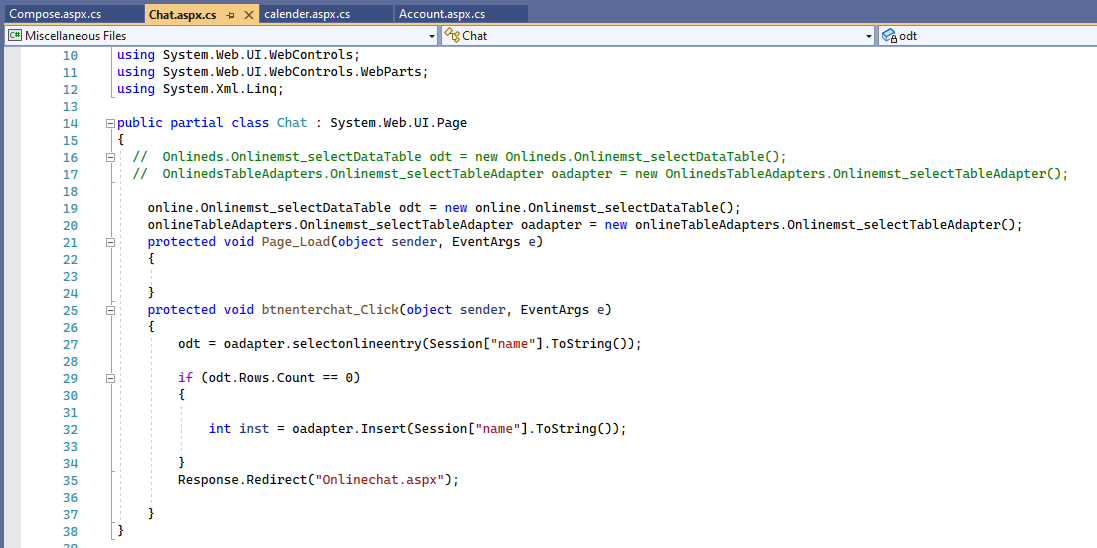
**A library project is present and classes are being created. - 2%**

The different classes have been created for the email management system. Like Account, calendar, chat, and compose mail. Below some screenshot has been put.

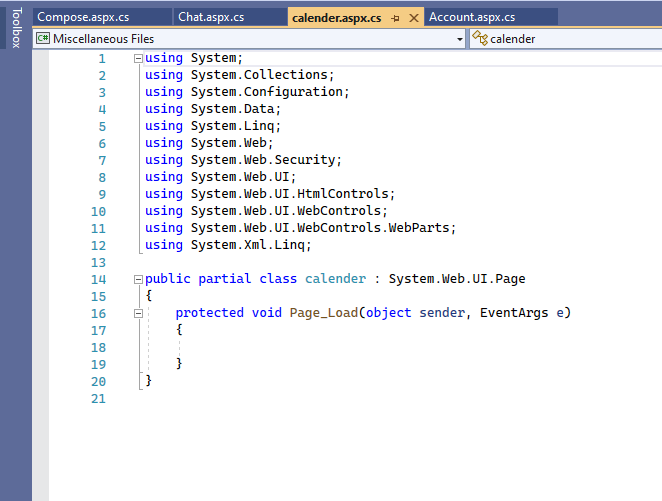
**Compose Email:**



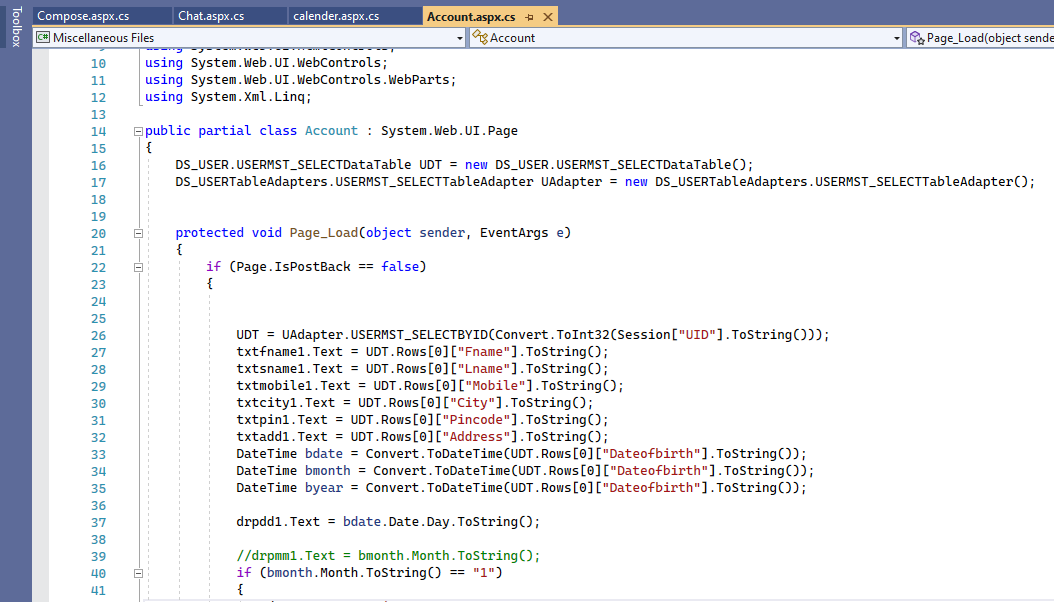
**Chat:**



**Calendar**



**Accounts:**

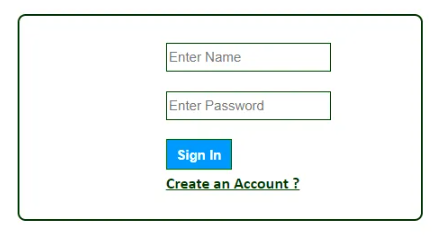


**User Interface**

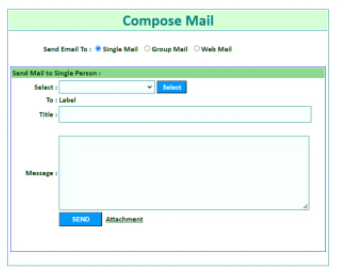
**Account Create**



**Password Set:**



**Compose Email:**



Working on other GUI as well.

1. **Registration**

**registration**

**2.Login**

**login**

**3 Drafts**

**drafts**

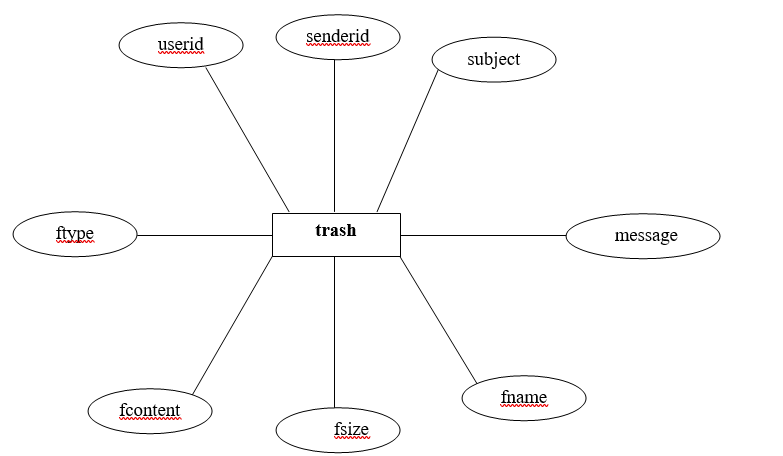
**4.Outbox**

**outbox**

**5 . Themes**

**tname**

**6. Trash**



**7. Composemail**

**composemail**

**Data Attributes**

* Login
* Drafts
* Outbox
* Themes
* Trash

**Integration Tests**

As we will design the project we will do the integration tests for created pages. The integration test for the login page would be.

* Checking the interface’s link in between login page and the home page, i.e. when a user registers in and inputs their password, they should be taken to the homepage.
* Checking the interface connection between the home page and the profile page, which should open up the profile page.
* Check the interface link between the network page and your connection pages; for example, pressing the welcome option on the system page's Invitation should display the received invitation in your connection page.
* Check the ui link between the Notifications pages and the Say Congrats button, which should take you to a new notification window when you click the Say Congrats toggle.