

1. ER Diagram:

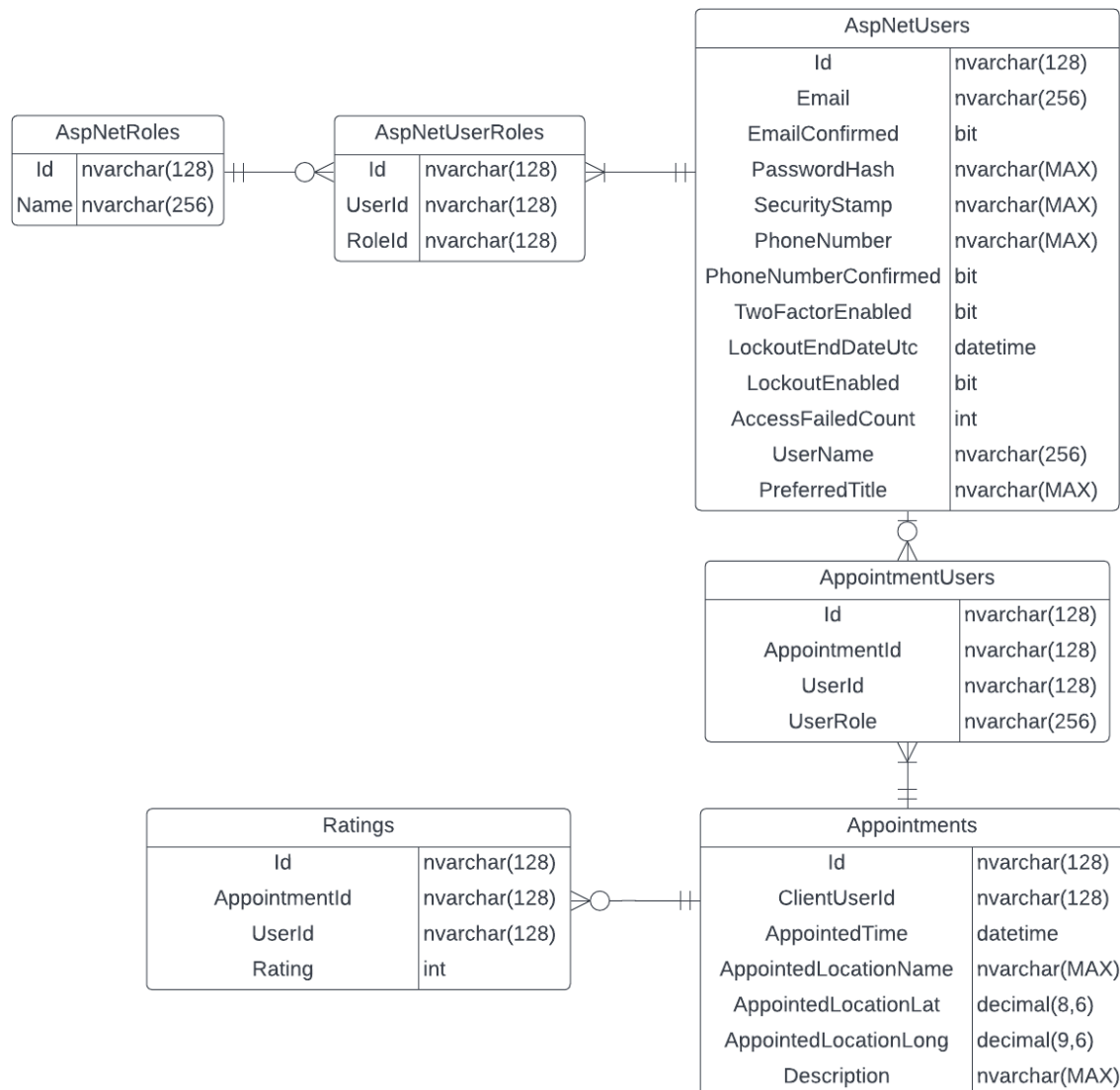


Figure 1: Entity Relation Diagram.

The ER Diagram above includes Business Requirements: B.2 (Date Storage), C.1 (Role-based Authentication), D4 (Booking Constraint) and E.2 (Geo Location, since I am storing latitude and longitude values of corresponding booking appointments).

Note that this ER diagram is still subject to change depending on new discoveries whilst I complete the implementation.

2. Implementation of Controller with user registration and authentication.

The default controller providing authentication in my current project is AccountController. Which simply handles the authentication via MS Identity.

Link:

https://github.com/3drdsh3in/FIT5032/blob/master/assignments/portfolio/FIT5032_Assignment_Portfolio/FIT5032_Assignment_Portfolio/Controllers/AccountController.cs

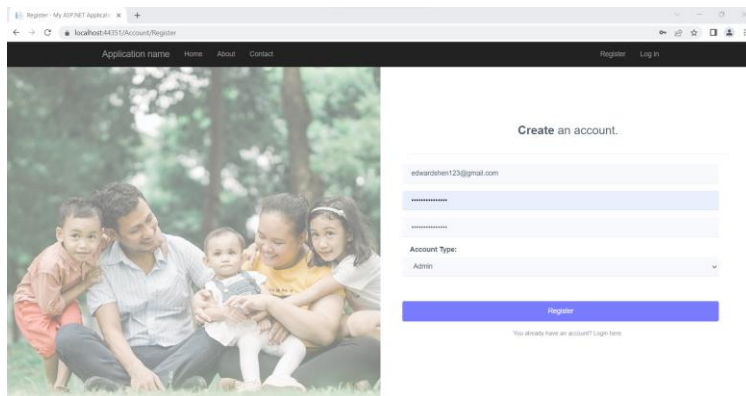
A screenshot of a web browser showing a registration form. The browser's address bar shows 'localhost:4431/Account/Register'. The page has a dark header with 'Application name' and links for 'Home', 'About', and 'Contact'. On the right, there are links for 'Register' and 'Log In'. The main content area features a large image of a family on the left and a registration form on the right. The form is titled 'Create an account.' and includes input fields for email (pre-filled with 'edwardshen123@gmail.com'), password, and confirm password. Below these is a dropdown menu for 'Account Type:' with 'Admin' selected. A blue 'Register' button is at the bottom of the form, followed by a link: 'You already have an account? Login here'.

Figure 2: Registration Form

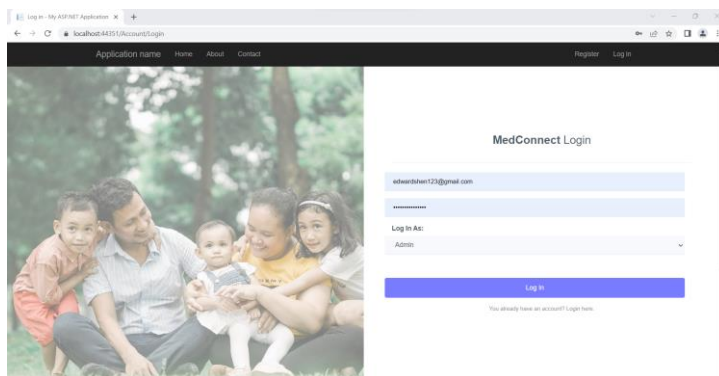
A screenshot of a web browser showing a login form. The browser's address bar shows 'localhost:4431/Account/Login'. The page has a dark header with 'Application name' and links for 'Home', 'About', and 'Contact'. On the right, there are links for 'Register' and 'Log In'. The main content area features a large image of a family on the left and a login form on the right. The form is titled 'MedConnect Login' and includes input fields for email (pre-filled with 'edwardshen123@gmail.com') and password. Below these is a dropdown menu for 'Log In As:' with 'Admin' selected. A blue 'Log In' button is at the bottom of the form, followed by a link: 'You already have an account? Login here'.

Figure 3: Logging In

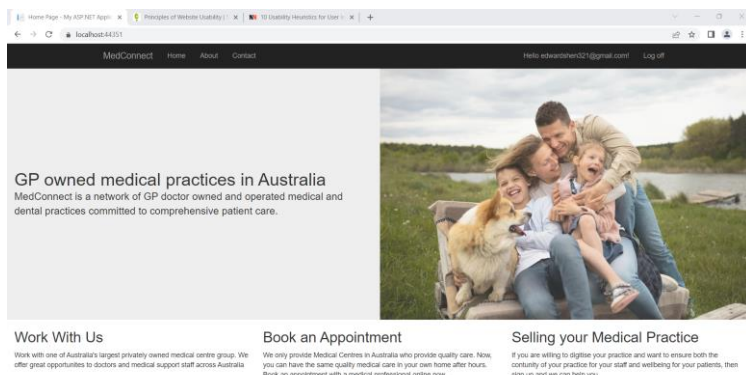
A screenshot of a web browser showing the home page of 'MedConnect'. The browser's address bar shows 'localhost:4431'. The page has a dark header with 'MedConnect' and links for 'Home', 'About', and 'Contact'. On the right, there is a user profile 'Hello edwardshen123@gmail.com!' and a 'Log off' link. The main content area is divided into two sections. The left section has a large image of a family and text: 'GP owned medical practices in Australia', 'MedConnect is a network of GP doctor owned and operated medical and dental practices committed to comprehensive patient care.', 'Work With Us', and 'We offer great opportunities to doctors and medical support staff across Australia'. The right section has a large image of a family and text: 'Book an Appointment', 'We only provide Medical Centres in Australia who provide quality care. Now, you can have the same quality medical care in your own home after hours. Book an appointment with a medical professional online now.', 'Selling your Medical Practice', and 'If you are willing to digitise your practice and want to ensure both the continuity of your practice for your staff and wellbeing for your patients, then sign up and we can help you.'.

Figure 4: Log in Screen upon clicking Create or Login.

3. Implementation of any other Controller and Usability evaluation

Implementation:

Link:

https://github.com/3drdsh3in/FIT5032/blob/master/assignments/portfolio/FIT5032_Assignment_Portfolio/FIT5032_Assignment_Portfolio/Controllers/AppointmentController.cs

Note that I'm not quite done with this controller. As of this submission only the Get Appointments Endpoint works to a standard, I deem barely acceptable (and even then, the UI still looks awkward). The Create Appointment Endpoint has been implemented in a primitive way to assist with testing the table functionality and will likely be changed to meet other business requirements down the line (Hence, please disregard it entirely for marking).

Usability Evaluation of the web application:

Appointment ID	Time	Location	Staff	Description
cadbfdf4-8328-4887-9bdc-51f6da61e7f7	16/09/2022	Monash Medical Centre	John Smith	Some Description that I'm going to make way too long just for testing sadassdasdkla haskdasjksd kladsjals jaskjklasdjaskd jklas dklasjd asdjasdklas klddsklsj

Figure 5: Table View Implementation for booking appointments.

The current version of the web application is far from ideal when considering it from a usability standpoint. Common principles of good website design such as consistency are not very well met due to a lack of responsiveness that has yet to be implemented (Haven't had the time to implement this yet). For example, the table view in Figure 5 (I think) is a bit awkward to read and digest. Without looking too hard we can easily notice misalignments of text being display vertically (The left edge of "MedConnect" title in the Navbar does not align with text beneath which makes it inconsistent with other parts of the website such as the Login/Registration screens where they do align). Additionally, it may be counterintuitive to display "Time" with a date and no actual timestamp (which was my original intention, ergo another element of poor usability I intend on improving). However, despite the current flaws it possesses I do still think that the current implementation at the very least still possesses some level of recognisability as it's not hard to tell that the page in Figure 5 we are showing is attempting to display an interactive table of appointments (with all the search features etc...). Overall, I believe I still have a long way to go with polishing the UI/UX if I intend on keeping the web app competitive at all from a usability standpoint.