COMP2245 - Project 2 (40 marks)

DUE DATE

Friday, December 12, 2025

Overview

For this project, you are tasked with creating an initial Minimum Viable Product (MVP) for a new Customer Relationship Management (CRM) System called **Dolphin CRM**. This CRM will allow users to add new contacts, view their details and add notes with information about interactions with the contact.

This project is a group project and will require you to be in groups of 3-5 people. You must declare your groups by the latest **Friday, November 14, 2025**.

Database

You should create a database called `dolphin_crm` which contains the following database tables with the following fields:

`Users` Table

Column Name	Data Type
`id` *(auto incrementing)*	INTEGER
`firstname`	VARCHAR
`lastname`	VARCHAR
`password`	VARCHAR
`email`	VARCHAR
`role`	VARCHAR
`created_at`	DATETIME

`Contacts` Table

Column Name	Data Type
`id` *(auto incrementing)*	INTEGER
`title`	VARCHAR
`firstname`	VARCHAR
`lastname`	VARCHAR
`email`	VARCHAR
`telephone`	VARCHAR
`company`	VARCHAR
`type` *(whether Sales Lead or Support)*	VARCHAR
`assigned_to` *(store the appropriate user id)*	INTEGER
`created_by` *(store the appropriate user id)*	INTEGER
`created_at`	DATETIME
`updated_at`	DATETIME

`Notes` Table

Column Name	Data Type
`id` *(auto incrementing)*	INTEGER
`contact_id`	INTEGER
`comment`	TEXT
`created_by` *(store the appropriate user id)*	INTEGER
`created_at`	DATETIME

NOTE

Note: You should create a file called `schema.sql` with the relevant `CREATE TABLE` statements for the tables above and include in your submitted code. Also, ensure you have an `INSERT` statement that adds a user with the email address `admin@project2.com` and the password `password123` (Of course, you should ensure that the password is hashed before using it in your `INSERT` statement).

Features

User Login

A user navigates to the login page and logs in with their *email address* and *password* (See *Figure 1*). The system keeps track of the user using PHP sessions. Once logged in, they are presented with the Dashboard/Home Screen (See *Figure 4*), which initially shows a list of all the contacts that have been added to the system.

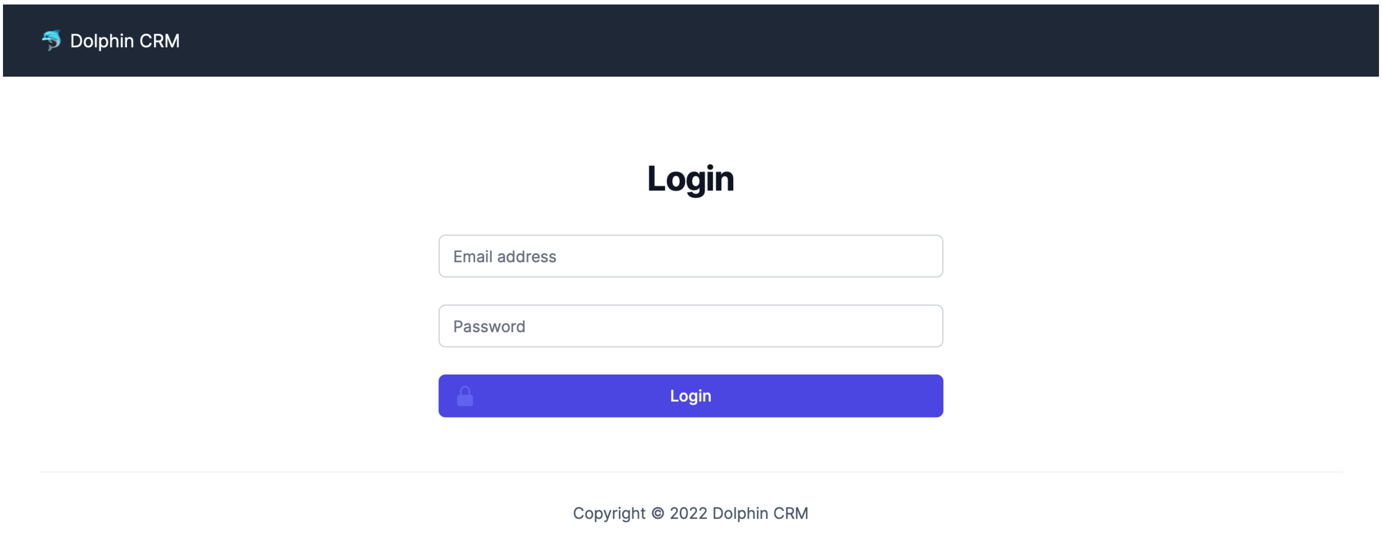

Figure 1: Dolphin CRM Login Screen

Adding a user

To simplify things for this assignment, users can only be added by an administrator; there is no feature for new users to self-sign up. An administrator logs in and completes the new user form (See *Figure 2*).

You should use regular expressions to ensure that passwords contain at least one number, one lowercase letter, one uppercase letter, and are at least eight (8) characters long. The password **MUST** be hashed before being stored in the database. I suggest you use the PHP `password_hash()` function to hash your passwords.

Users can have a role of either *Admin* or *Member*.

Ensure that all input fields are validated and that user inputs are escaped and sanitized before being inserted into the database. Also, provide the user with feedback if the new contact was added successfully or if there are any errors with their submission.

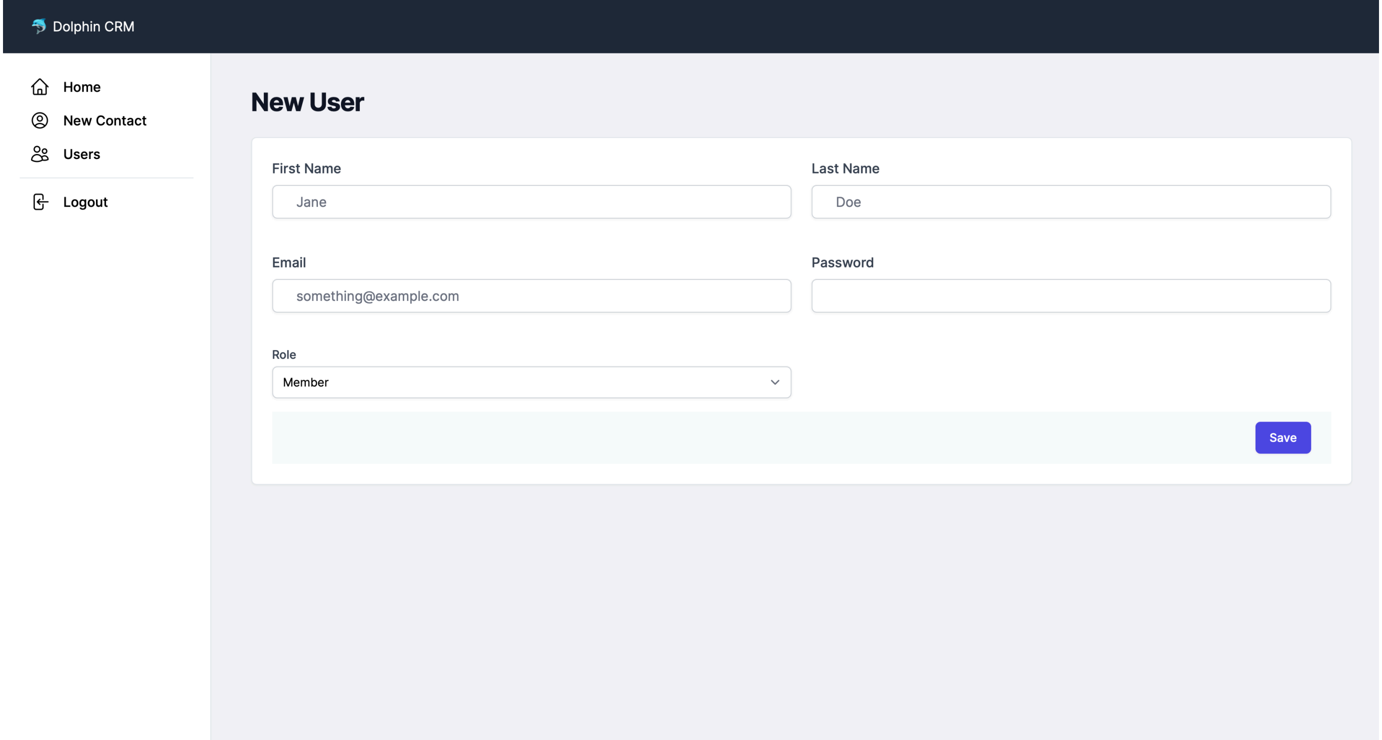

Figure 2: Wireframe of New User creation screen

View a list of users

The system should also have a page that lists all the users of the system. It should show their full name, email, role and date created in a table (See *Figure 3*). Only an *Admin* should be able to view the list of all users.

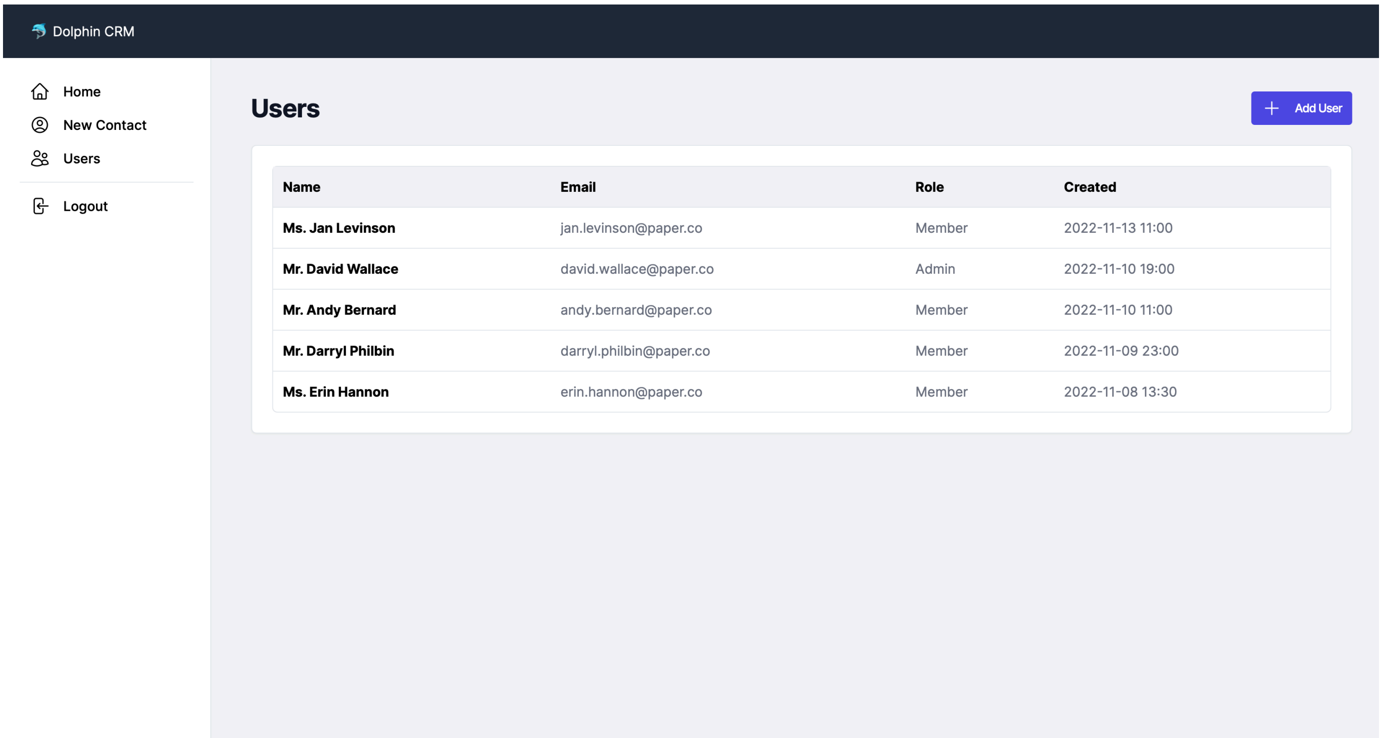

Figure 3: List of Users

User Logout

There will be a logout link/button that a user can click to log out of the system. When this is done, the system should destroy the PHP Session and redirect the user to the login screen.

Dashboard/Home Screen

The Dashboard/home screen allows a logged-in user to view a list of all contacts added to the system. The list of contacts should display the contact's title (e.g., Mr., Mrs., Ms., Dr., or Prof.) and full name, Email, Company, and Type of Contact (e.g., Sales Lead or Support). There should also be a link that, when clicked, will allow the user to view the full details of the contact. (See *Figure 4*)

The page should also have a list of filters to display "*All Contacts*", no matter their type, "*Sales Leads*" only, "*Support*" only or contacts "*Assigned to me*", which should display only the contacts assigned to the currently logged in user.

There should be a call to action button to *Add New Contact* in the top right of the Dashboard/Home screen.

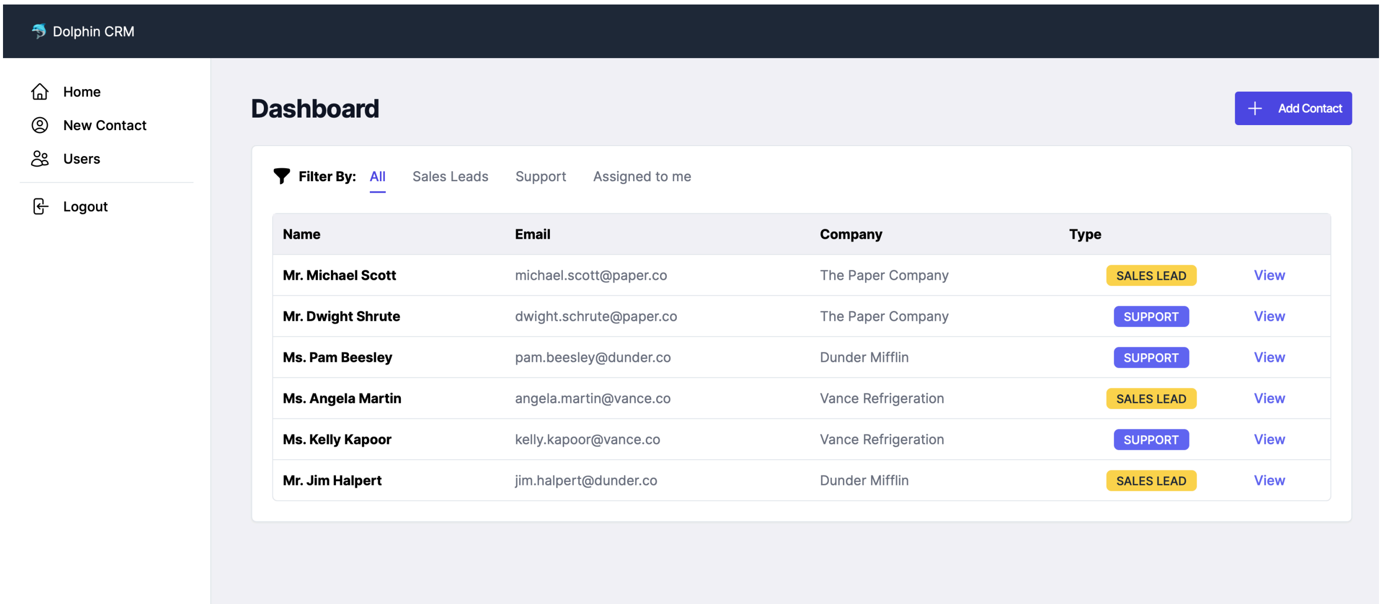

Figure 4: Dolphin CRM Dashboard

Create a New Contact

The Create a New Contact screen includes a form with the following input fields: **Title**, **First Name**, **Last Name**, **Email**, **Telephone**, **Company**, **Type** (e.g. *Sales Lead* or *Support*) and **Assigned To** (*which should list the names of all the users in the system*) input fields. (See *Figure 5*).

Whenever a new contact is created, you should also store the user ID of the user who created the contact, and the `created_at` and `updated_at` date columns should be set to the current date and time.

Ensure that all input fields are validated and that user inputs are escaped and sanitized before being inserted into the database. Also, provide the user with feedback if the new contact was added successfully or if there are any errors with their submission.

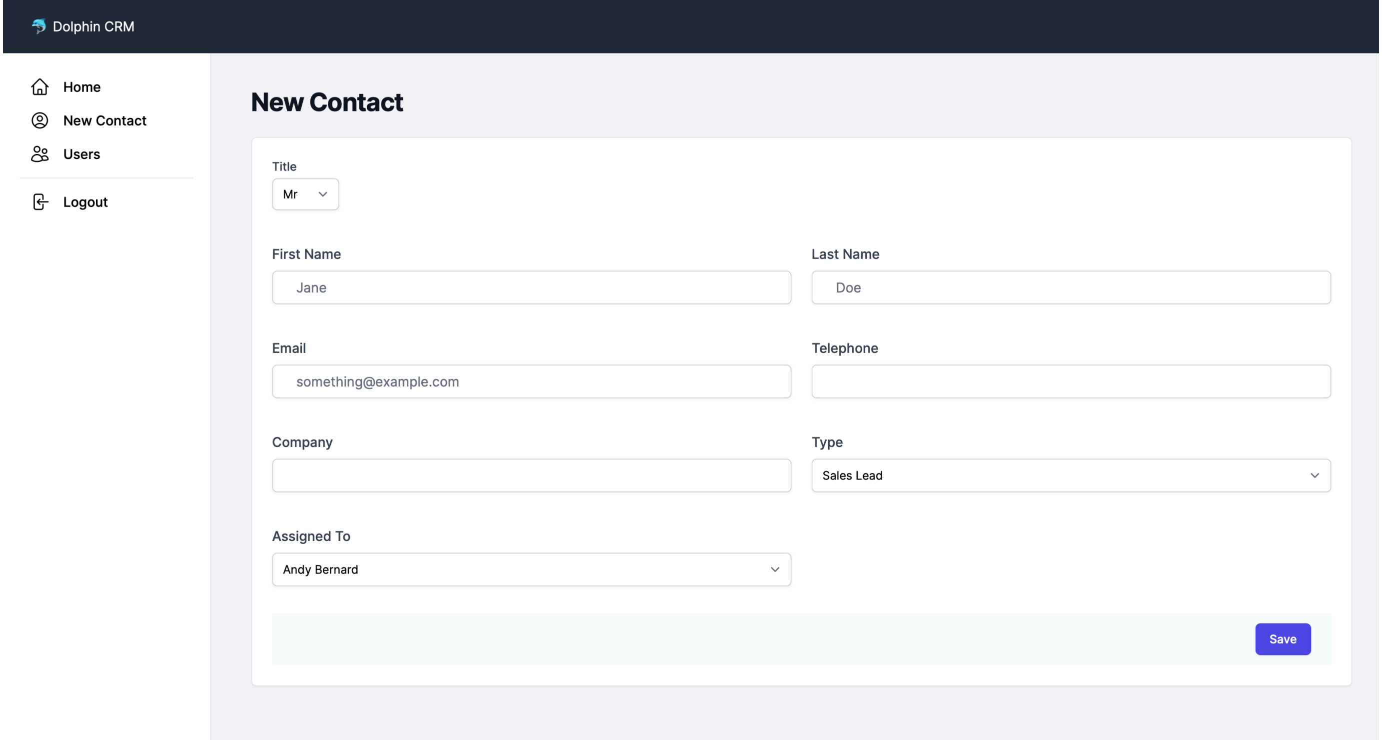

Figure 5: Wireframe of new Contact page

Viewing Full Contact Details

When the name of a contact or the *View* link on the Dashboard/Home screen is clicked, it will open and show the full details of that contact (See *Figure 6*). These details should include the *Title* and *Full Name*, *Email*, *Company*, *Telephone*, *Date Created and by whom*, *Date last updated*, as well as who the contact is *assigned to*.

In addition to viewing the full contact details, a user should be able to assign the Contact to themself by selecting the "Assign to me" button. A user can also switch the type of the contact to a *Sales Lead* if it is currently of the type *Support* or switch to *Support* if it is currently of the type *Sales Lead*. When either of these options is changed, you should also update the `updated_at` timestamp on the contact.

A user should also see a list of notes for that particular contact (See *Figure 7*). Each note should include the name of the user who created the note, the actual comment, and the date the note was added. The user should also be able to add a new note for the contact.

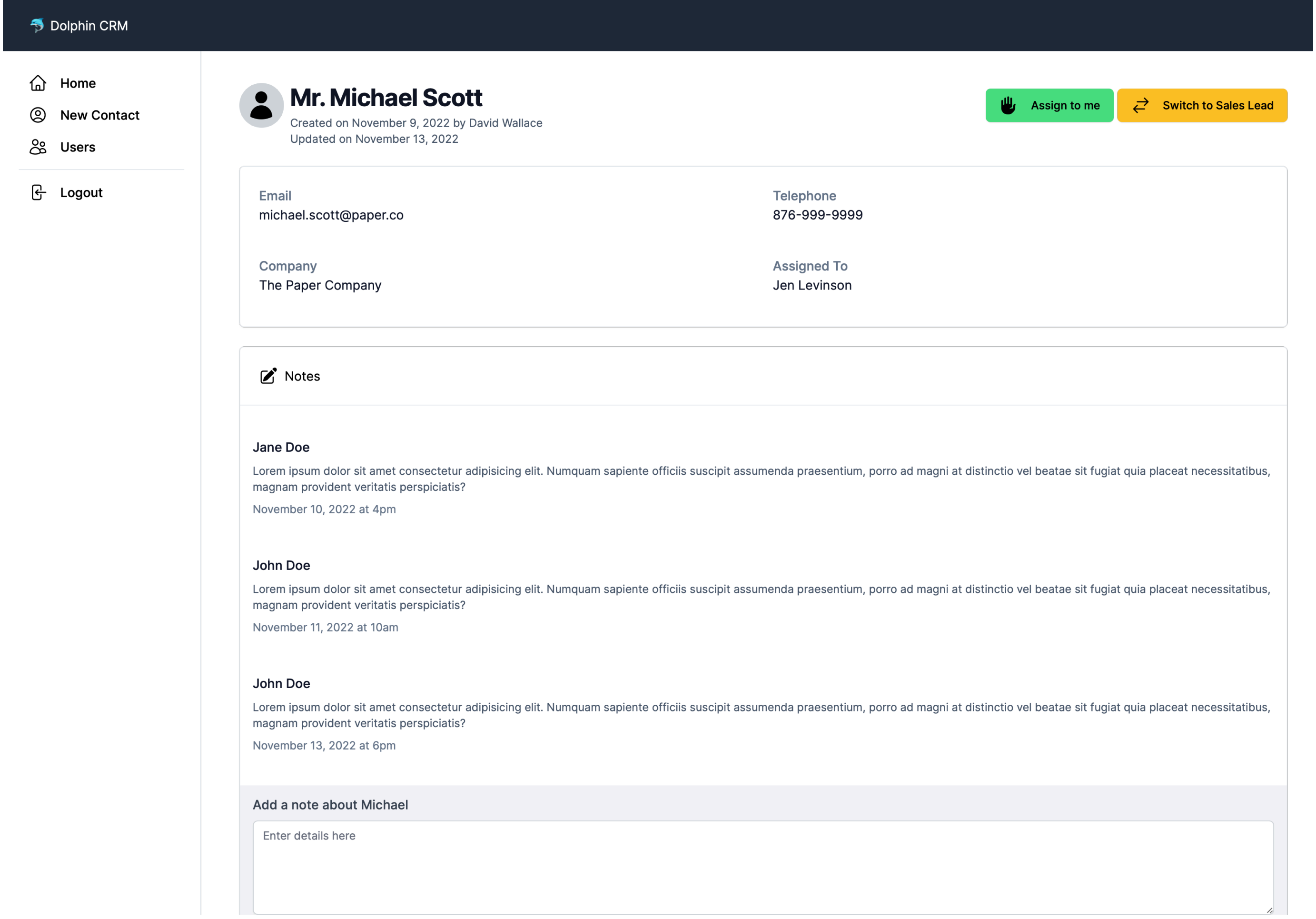

Figure 6: Wireframe of Contact details page

Adding a note to the Contact

A user should be able to add a note to a contact to share internal information, log phone calls, or document conversations with their contacts.

When viewing a contact, a section should be displayed below the contact details, showing existing notes for the contact, as well as a field to enter a note about the contact and a button to save the note (See *Figure 7*). When a new note is added, the `updated_at` column on the contact should also be updated.

Ensure that all input fields are validated and that user input is escaped and sanitised.

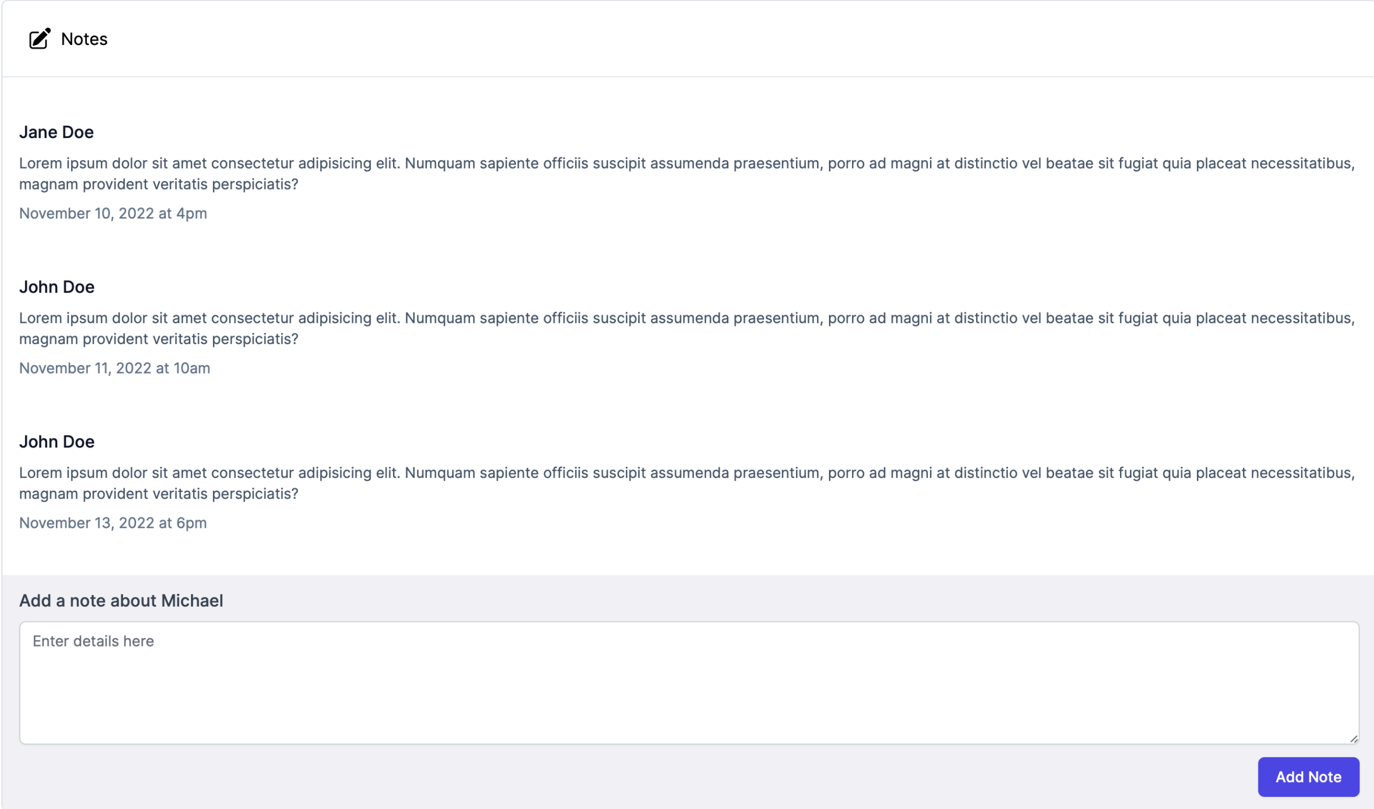

Figure 7: Wireframe of Notes section on Contact Details page

No Page Refreshes by using AJAX

You will need to implement an AJAX-based approach to loading new content or updating existing content in the browser, as well as for form submissions. In other words, all pages should load/submit the appropriate data without requiring the browser to refresh. Utilize your knowledge on the `XMLHttpRequest` object, jQuery `.ajax()` function or the `fetch()` method from the Fetch API to make your AJAX requests.

Future Improvements

In the next development cycle, we hope to allow Dolphin CRM to include other features such as the ability to edit and delete users and contacts, add Tasks, upload files, send emails directly to Contacts and add Cases, which can help us to organise our communication – notes, emails and files together. However, for now, that is not necessary for our initial MVP.

Submission

You will submit using the "Final Project Submission" link on the course platform. Since it is a group project, only one group member is required to submit the relevant project links.

You are required to commit your code to GitHub and submit the link to your group project GitHub repository (e.g. https://github.com/yourusername/comp2245-finalproject).