*CA2:*

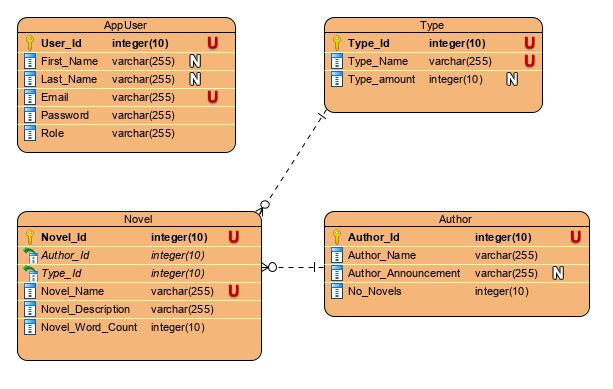
In this CA, I choose to develop a novel website for the application area. The application will have a home page which have a table with list of novels that’s order by its id.

And you can filter it by its author using the nav bar at the left-hand side with the name of each author. Or you can use the side bar below the author names to filter the novels by its type.

The functionality of the website is to see all the novels and manipulate it by different type of users. There are 4 types of users: unregistered, registered(normal), manager, and admin.

* The users that’s unregistered, they can only see the novel table and filter the novels by authors or types. But the normal user who’s registered can see all the novels and likes novels (click the like button to increase the number of likes that a novel gets from all the users). Only the registered user can see the like button when they login.
* The manager users can add novel by the add button that appears at the bottom of the webpage. And the manager can update and delete a novel by the edit and delete buttons that appears at the end of each row in the table. The add, edit, delete button will only appear when the user login as a manager.
* The admin user can see all the novels but can’t make any changes to the novels. The admin can manipulate all the users that’s registered. Admin can add or delete users (normal user or manager) including themselves. So, they can delete their account or delete other admin’s account. They will have a user webpage when they login. So they can see all the users and edit them by the add, edit and delete buttons.

For the database, I will create 4 tables that stores the information of the novels, authors, types and users. The ERD below will show all the attributes that each table contain:



* + For the Novel table, it has another attribute:
    - Novel\_Like integer (10)
  + The relationship between author and novel is one to many relationships. So, one author can write many novels. And the relationship between type and novel is one to many as well, that’s because one type can have many novels.
* The api endpoint required is:
  + http:localhost:8080/ - to show the table with the list of novels as HTML.
  + http:localhost:8080/novel - to show the list of novels in JSON.
  + http:localhost:8080/appUser - to show all the users and user details in JOSN.
  + http:localhost:8080/login - uses for the users to login.
* Three types of users:
  + User(normal) - can see all the novels, but they can click the like button for the novels so all the users can see how many people liked a novel (which novel are most populous).
  + Manager - can add, update,delete all the novels and authors.
  + Admin - can edit the details of all the users and have all the functionality that manager has.