DESIGN NOTES : PROJECT 2 SIT 102

To design such a program ,we first need to know the basic approach of the code.

We can start by understanding the approach of the program which in this case is managing a database of movies and their reviews. Now we can have a look at the structure of the code which clearly will include components like structs, enums, functions and loops.

These functions each have a separate purpose, For example, addMovie() adds a movie to the database, removeMovie() removes a movie by name, displayAllMovies() shows all movies, and so on.

During the flow of the program it can be seen that how user input is collected, menus are displayed, and actions are performed based on user choices.

I would like to take a deeper look into the designing of the program.

Modular structure:

The program is structured into separate functions and structs for managing movies and reviews, displaying menus, and handling user input. This modular approach makes the code easier to understand, maintain, and extend.

Object oriented design:

The use of structs for Movie, Review, and Database reflects an object-oriented design, where each struct represents a real-world entity with its attributes and behaviors. This design simplifies the management of movie data and operations.

Created a user interface :

The program provides a clear and intuitive user interface with a main menu and genre selection menu. Each menu option corresponds to a specific action, making it easy for users to interact with the program.

Making sure the program does not crash if an unexpected value is entered, instead display a simple message :

For instance, if a user tries to remove a movie that doesn't exist in the database, the program provides feedback instead of crashing or malfunctioning.

Can be used again and can get more features included:

We can use the same bits of this program in some other programs as well and some more features like sorting movies by date or searching for a specific director’s movie can be added.