

Patrones utilizados

1.- Singleton

Utilice singleton para tener un punto de acceso a algunas funciones, como las de validación de los usuarios, pero sobre todo a las clases de los alumnos profesores etc.

Para eso ocupé una clase Data que se creaba y leía una vez, y para optimizar eso el singleton con su única instancia era la mejor opción.

2.- Builder

Utilice builder para crear las diferentes GUI. Una para alumno, que permite descargar, ver calificaciones e info, la de profesor que no permite descargar y la de supervisor que solo dejaba ver info básica. Con esto en mente, usar un builder implementando estas 3 funciones o las que fueren necesarias respecto a que usuario está usando la interfaz fue la mejor opción, ya que me permite solo construir las opciones que necesito.

Las clases usadas fueron.

Supervisor, Alumno y Profesor para los usuarios, donde tanto profesor como alumno heredan de supervisor debido a los valores que comparten.

Data y Singleton para aplicar el patrón singleton.

Director, Builder, AlumnBuilder, SupervisorBuilder y TeacherBuilder.

En este caso el producto es un Form llamado UserForm, personalizado dependiendo el usuari..

Además de esto un form llamado Login para el acceso y validación de usuarios.