

# Recomendador

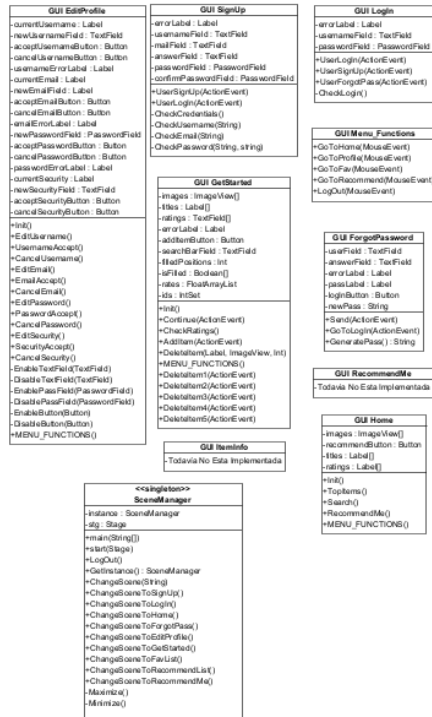
Diagrama de Clases

**Gerard Madrid Miró**  
**Guillem Gràcia Andreu**  
**Ismael Quiñones Gama**  
**Pol Ken Galceran Kimura**

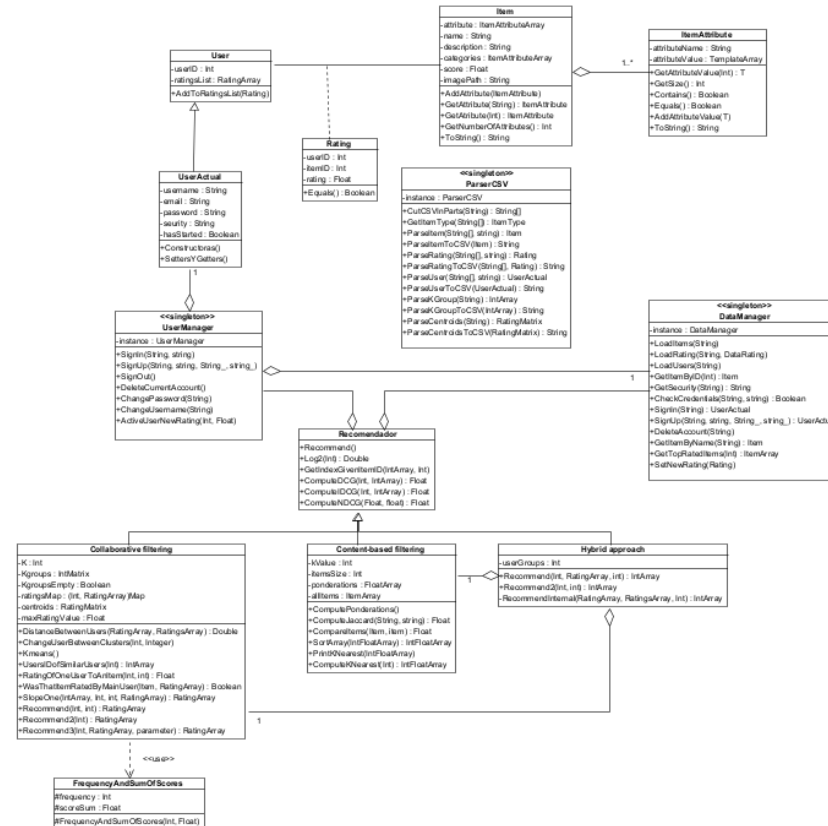
# Diagrama de clases general

Dado que el diagrama tiene un tamaño bastante grande (imagen en mejor calidad en DiagramaDeClases.jpg), hemos preferido separarlo en las capas para que se pueda entender mejor. Aún así, siempre está bien tener una visión general:

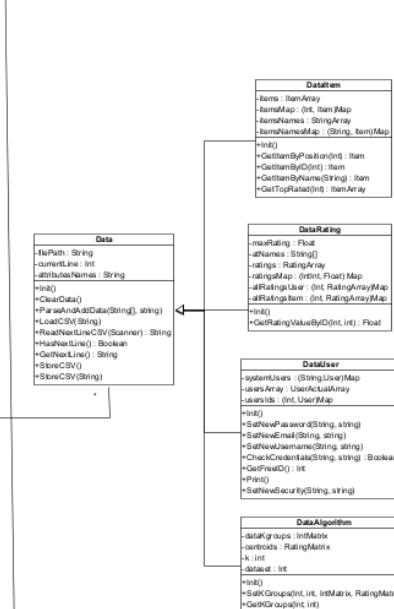
## PRESENTATION LAYER



## DOMAIN LAYER



## PERSISTENCE LAYER



# Capa de Presentación

```

classDiagram
    class GUI_EditProfile {
        -currentUsername : Label
        -newUsernameField : TextField
        -acceptUsernameButton : Button
        -cancelUsernameButton : Button
        -usernameErrorLabel : Label
        -currentEmail : Label
        -newEmailField : Label
        -acceptEmailButton : Button
        -cancelEmailButton : Button
        -emailErrorLabel : Label
        -newPasswordField : PasswordField
        -acceptPasswordButton : Button
        -cancelPasswordButton : Button
        -passwordErrorLabel : Label
        -currentSecurity : Label
        -newSecurityField : TextField
        -acceptSecurityButton : Button
        -cancelSecurityButton : Button
    }

    class GUI_SignUp {
        -errorLabel : Label
        -usernameField : TextField
        -mailField : TextField
        -answerField : TextField
        -passwordField : PasswordField
        -confirmPasswordField : PasswordField
        +UserSignUp(ActionEvent)
        +UserLogin(ActionEvent)
        -CheckCredential(s)
        -CheckUsername(String)
        -CheckEmail(String)
        -CheckPassword(String, string)
    }

    class GUI_Login {
        -errorLabel : Label
        -usernameField : TextField
        -passwordField : PasswordField
        +UserLogin(ActionEvent)
        +UserSignUp(ActionEvent)
        +UserForgotPass(ActionEvent)
        -CheckLogin()
    }

    class GUI_Menu_Functions {
        +GoToHome(MouseEvent)
        +GoToProfile(MouseEvent)
        +GoToFav(MouseEvent)
        +GoToRecommend(MouseEvent)
        +Logout(MouseEvent)
    }

    class GUI_RecommendMe {
        -saveListButton : Button
        -moreRecommendButton : Button
        -image1 : ImageView
        -image2 : ImageView
        -image3 : ImageView
        -image4 : ImageView
        -image5 : ImageView
        -title1 : Label
        -title2 : Label
        -title3 : Label
        -title4 : Label
        -title5 : Label
        -feedbackLabel : Label
        -errorLabel : Label
        -list : IntArray
        -currentRecommendation : int
    }

    class GUI_ForgotPassword {
        -userField : TextField
        -answerField : TextField
        -errorLabel : Label
        -passLabel : Label
        -loginButton : Button
        -newPass : String
        +Send(ActionEvent)
        +GoToLogin(ActionEvent)
        +GeneratePass() : String
    }

    class GUI_Home {
        -images : ImageView[]
        -recommendButton : Button
        -titles : Label[]
        -ratings : Label[]
    }

    class GUI_GetStarted {
        -images : ImageView[]
        -titles : Label[]
        -ratings : TextField[]
        -errorLabel : Label
        -addItemButton : Button
        -searchBarField : TextField
        -filledPositions : Int
        -isFilled : Boolean[]
        -rates : FloatArrayList
        -ids : IntSet
    }

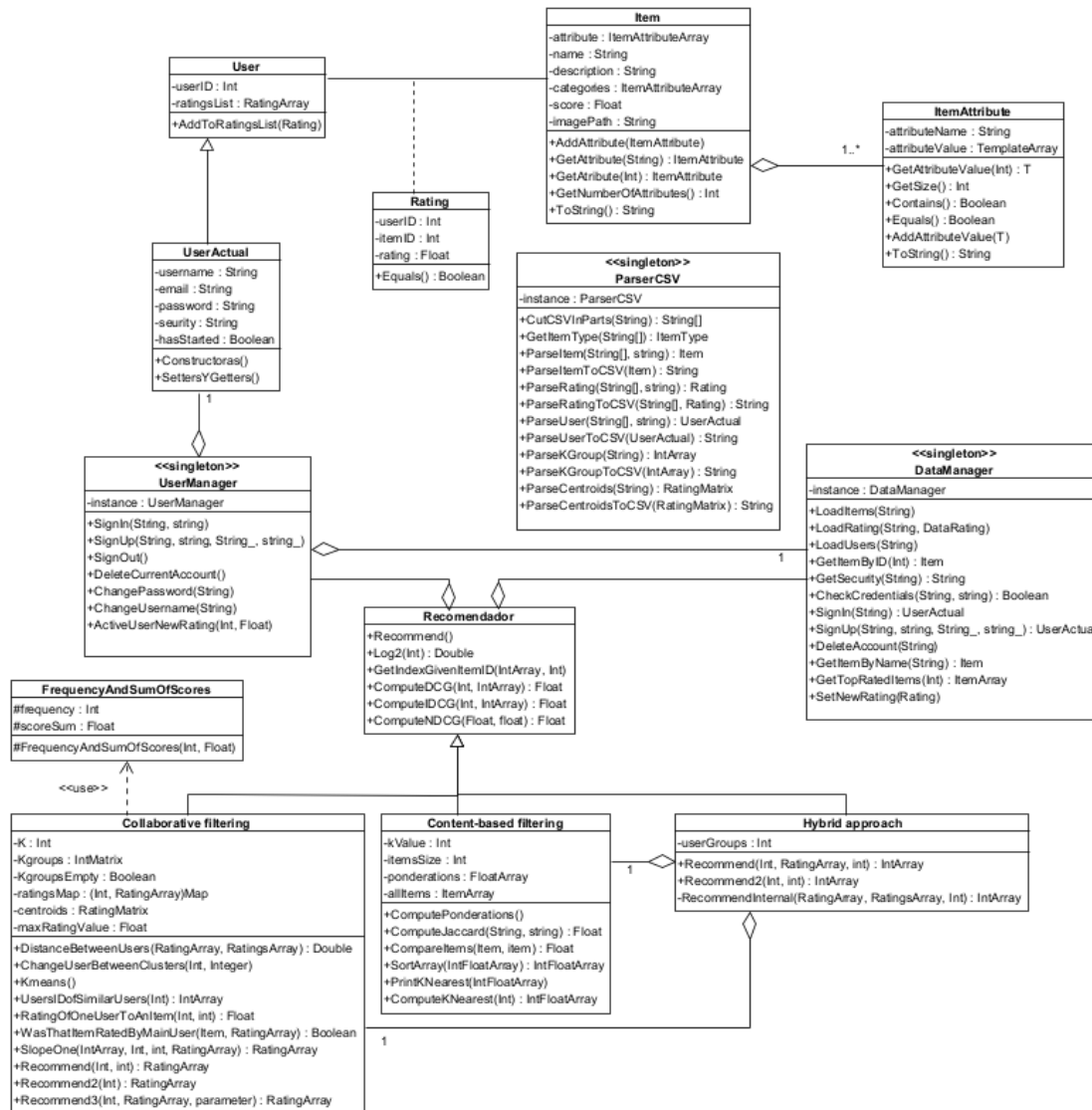
    class Singleton_SceneManager {
        <<singleton>>
        -instance : SceneManager
        -stg : Stage
        +main(String[])
        +start(Stage)
        +Logout()
        +GetInstance() : SceneManager
        +ChangeScene(String)
        +ChangeSceneToSignUp()
        +ChangeSceneToLogin()
        +ChangeSceneToHome()
        +ChangeSceneToForgotPass()
        +ChangeSceneToEditProfile()
        +ChangeSceneToGetStarted()
        +ChangeSceneToFavList()
        +ChangeSceneToRecommendList()
        +ChangeSceneToRecommendMe()
        -Maximize()
        -Minimize()
    }

    GUI_EditProfile -- GUI_SignUp
    GUI_EditProfile -- GUI_Login
    GUI_EditProfile -- GUI_Menu_Functions
    GUI_EditProfile -- GUI_RecommendMe
    GUI_EditProfile -- GUI_ForgotPassword
    GUI_EditProfile -- GUI_Home
    GUI_EditProfile -- GUI_GetStarted
    Singleton_SceneManager -- GUI_EditProfile
    Singleton_SceneManager -- GUI_SignUp
    Singleton_SceneManager -- GUI_Login
    Singleton_SceneManager -- GUI_Menu_Functions
    Singleton_SceneManager -- GUI_RecommendMe
    Singleton_SceneManager -- GUI_ForgotPassword
    Singleton_SceneManager -- GUI_Home
    Singleton_SceneManager -- GUI_GetStarted
  
```

The diagram illustrates the architecture of a movie recommendation application. It features several GUI components and a central Singleton SceneManager.

- GUI EditProfile**: Manages user profile information, including username, email, and password fields, along with buttons for saving and canceling changes.
- GUI SignUp**: Handles user registration, including username, email, and password validation.
- GUI Login**: Manages user login, including username, password, and forgot password functionality.
- GUI Menu\_Functions**: Provides navigation between different parts of the application (Home, Profile, Favorites, Recommendations, Logout).
- GUI RecommendMe**: Displays movie recommendations, including image views, titles, ratings, and a feedback label.
- GUI ForgotPassword**: Handles password recovery, including user identification, password generation, and login redirection.
- GUI Home**: Displays a list of movies with image views, titles, and ratings.
- GUI GetStarted**: Provides a guided tour or initial setup for new users, including image views, titles, ratings, and a search bar.
- Singleton SceneManager**: Manages the application's state, including the current scene, stage, and navigation between different GUI components.

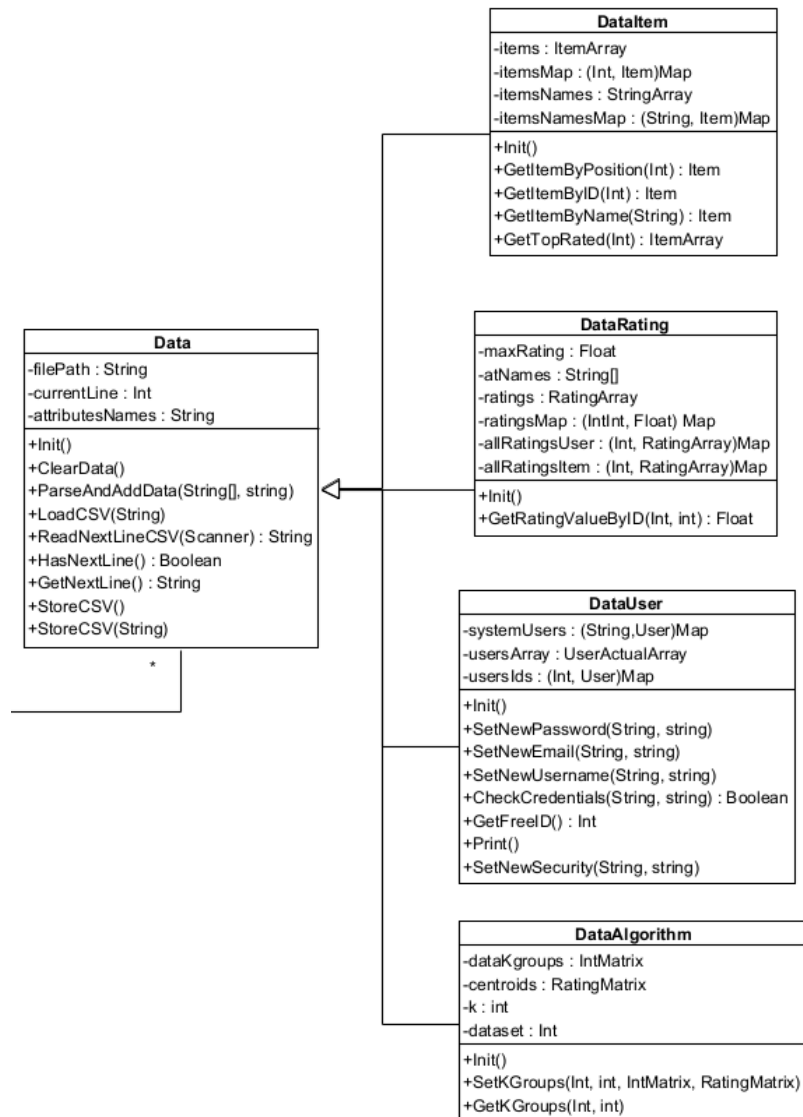
# Capa de Dominio



En la capa de datos, encontramos todas las clases encargadas de gestionar el sistema en general. Podemos encontrar los items y sus atributos, los usuarios (y el Usuario Actual activo), el parser que gestiona la lectura de datos, y los dos puentes entre las capas vecinas: **UserManager** hará de puente entre Dominio y Presentación, mientras que **DataManager** hará de puente entre Dominio y Persistencia.

Podemos encontrar también el recomendador, los tres tipos que heredan de él, y como el Hybrid Approach contiene una instancia de ambos Collaborative Filtering y Content Based Filtering.

# Capa de Persistencia



En la capa de persistencia encontramos la clase abstracta Data. Esta clase gestiona todos los datos y es padre de las 4 clases de datos principales de nuestro sistema. Estas 4 clases son:

**DataItem:** Gestiona todos los datos referentes a los Ítems y sus atributos.

**DataRating:** Gestiona todos los datos referentes a los Ratings (valoraciones) y los usuarios que las dan.

**DataUser:** Gestiona todos los datos referentes a los usuarios, incluyendo sus credenciales.

**DataAlgorithm:** Gestiona todos los datos referentes a los algoritmos para evitar tener que hacer cálculos innecesarios que podrían ser almacenados.