Suppose that your system has the following Use Case Description and Actors:

Actor	Use Case
Scriptwriter	Writing a Spectacle – Consists of creating a spectacle and spectacle scene. We also give a name to the spectacle and write a plot.
	Create Character – Create a character and connect to the spectacle
Programmer	Create Program - Create a show supported by the spectacle.
Director	Create show scenes – He decides where each scene begins and ends.
Artist	Entrance_Exit - Each time that enters or exists on the stage, the artist registers its entrance, also identifying the role (character) he is playing.
Spectator	See the program – The program is a list of all the shows of a specific Spectacle
	See Spectacle Information – See generic information about the show, its scenes
	See Artist Information – see list of artists that are on the stage
	See Character Information – See a list of characters that are in the scene

Consider the following relational scheme:

Spectacle(IdSpectacle, Title)

SpectacleScene(*IdSpectacle, SceneNumber,* Plot)

Show(*IdShow, IdSpectacle,* Date, Time)

ShowScene(*IdShow,SceneNumber,*B_time,E_time*)

Entrance_Exit(IdPerfomance, *IdShow,SceneNumber, IdCharacter, IdArtist,* Data, Time, Type{Entrance, Exist})

Character(IdCharacter,Gender, Name, Descrition, *IdSpectacle*)

Artist(IdArtist, Name, Experience, Description)

You are asked to implement a solution supported in MS-ACCESS. To implement, you should pursue the following steps:

1. Database (9 points)
- Create the tables according to the relational scheme.
- Connect the tables according to the relational scheme.

2. Queries (4 points)
Create the queries corresponding to the use cases related to the spectator.

3. Forms (3 points)
- Create all the forms corresponding to each one of the use cases of the system

4. Menu (2 points)
- Create the following menus: Scriptwriter Menu, Programmer Menu, Programmer Menu, Director Menu, Artist Menu and Spectator Menu.

4. Improvements (2 points)
- Add improvements according to what you consider adequate