

# Introduction

---



## A1.1 Learning Activity

Requirements Modeling through UML



### Instructions

- Based on the provided document by the teacher and other sources, develop the indicated diagrams in the development for the case study that has been reviewing during the subject.
- Use the same tools from past challenges, such as drawio, or others that allow you to create the diagrams or images that can illustrate the concept.
- The team will explain what does each type of diagram represents, what elements are part of it, as well as its interpretation within the case study.
- Every activity or challenge must be realized using **Markdown style with .md extension** and VSCode development environment, or another platform can be used, such as **Norton**, it has to be as a **single page** document, which means if the document has images, links o another external document it must be accesed from tags and links, and must be named using the nomenclature **A1.1\_ActivityName\_Student'sName.pdf**.
- As a requirement the .md file must contain a tag of the link to the repository of your Github document, for example a **Link to my GitHub** and after concluding the assignment it must be submitted to Github.
- From the **.md** file export a **.pdf** that must be uploaded to classroom in the corresponding section, serving as evidence of turned in, since it being the **official** platform, here you will receive the activity's result.
- Considering that the .pdf file was obtained from the .md file, both must be identical.
- Your repository, besides containing a **readme.md** inside the root directory containing student's information, team, subject, career, teacher's information, and also a logo or pictures, must contain a section of contents or index, which are **links to your md documents**, try not to use text to indicate internal or external links.
- It's suggested to use a structure similar to the one indicated below, however, you can use any other structure that can help you organize your repository.

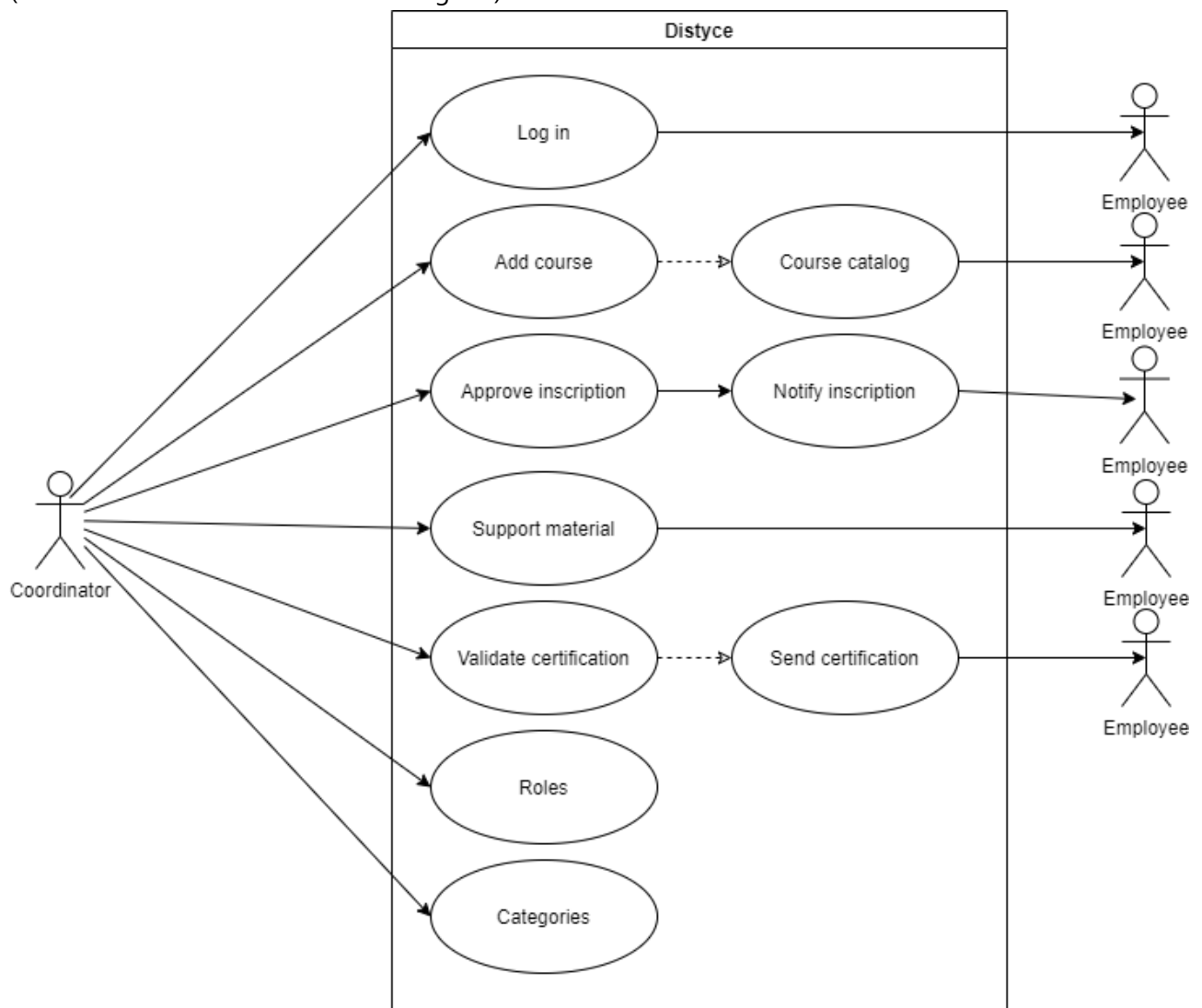
```
| readme.md
| | blog
| | | Cx.1_NombredelaActividad.md
| | | Ax.1_NombredelaActividad.md
| | diagrams
| | docs
| | html
```

| | img  
| | pdf

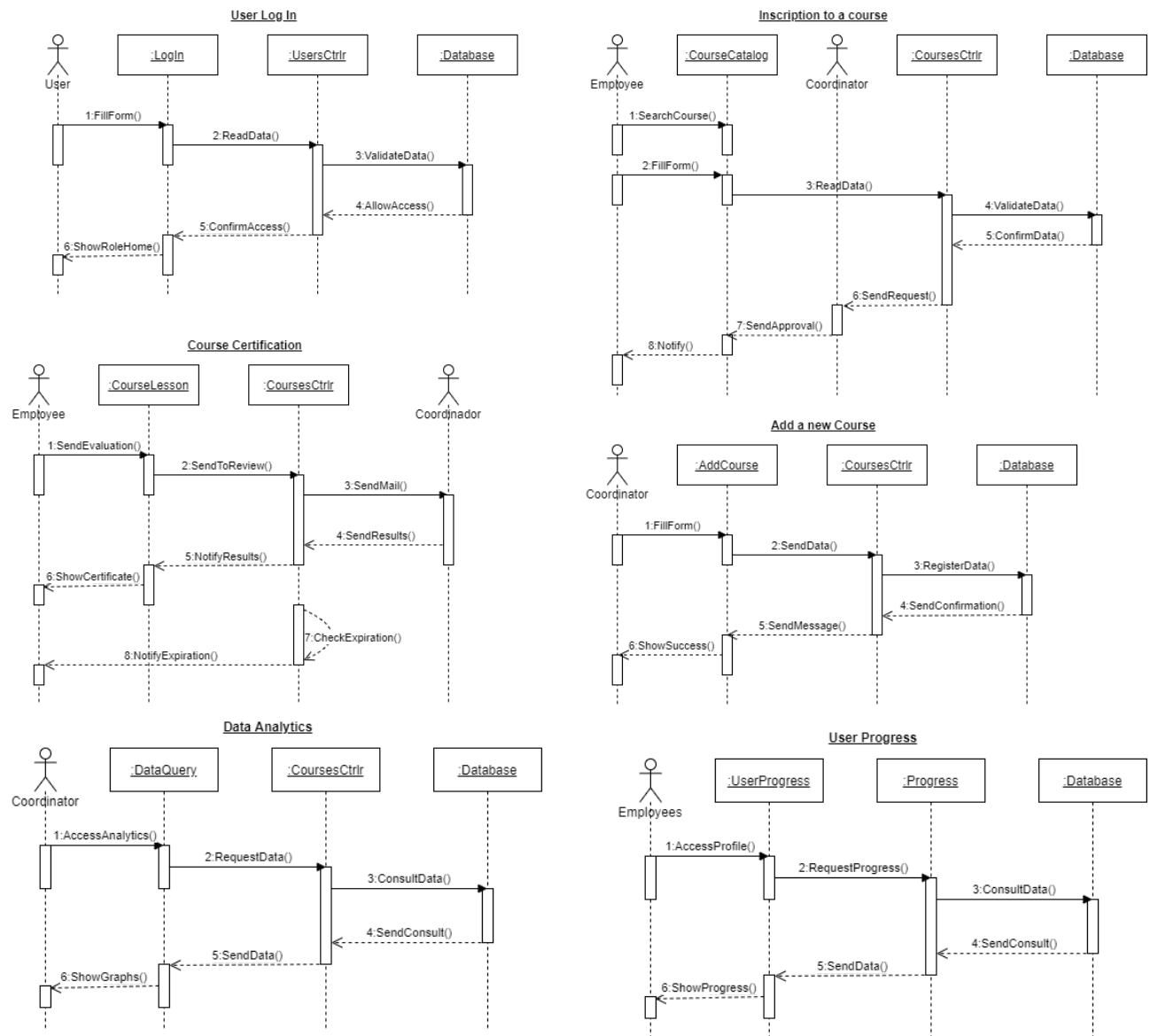


## Development

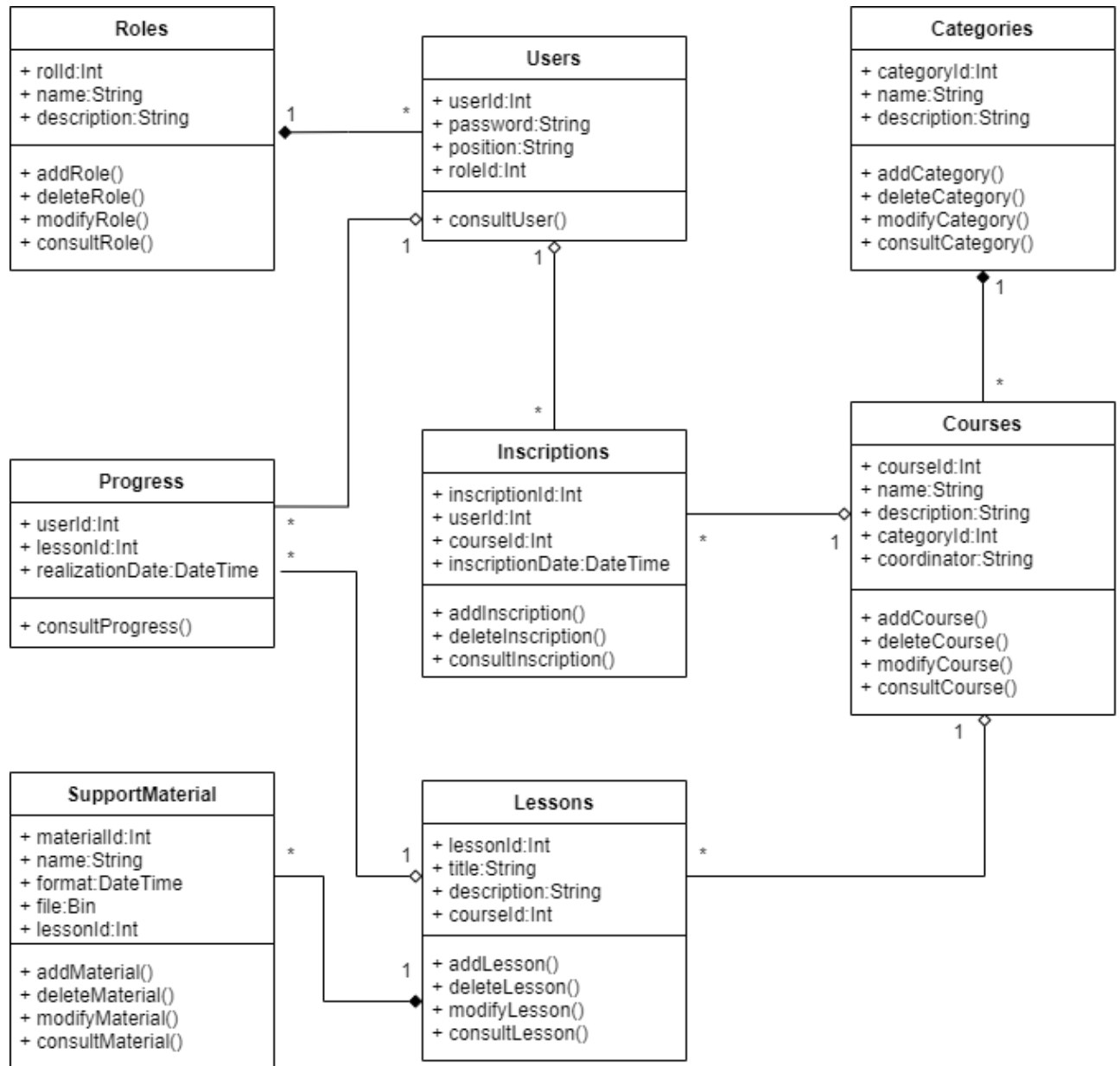
- Based on each one of the previously established user and system requirements for the case study, draw up the requested modeling diagrams:
- A diagram containing the relationships between the different use cases, only for the scenario in which the user interacts with the system, using the main or central functions according to each case study. (Include at least 5 elements of the diagram)



- The sequence diagram that explains the communication between the different GUIs for the use cases of the previous point, considering that the user has successfully accessed into the system. (Include at least 5 elements from the diagram)

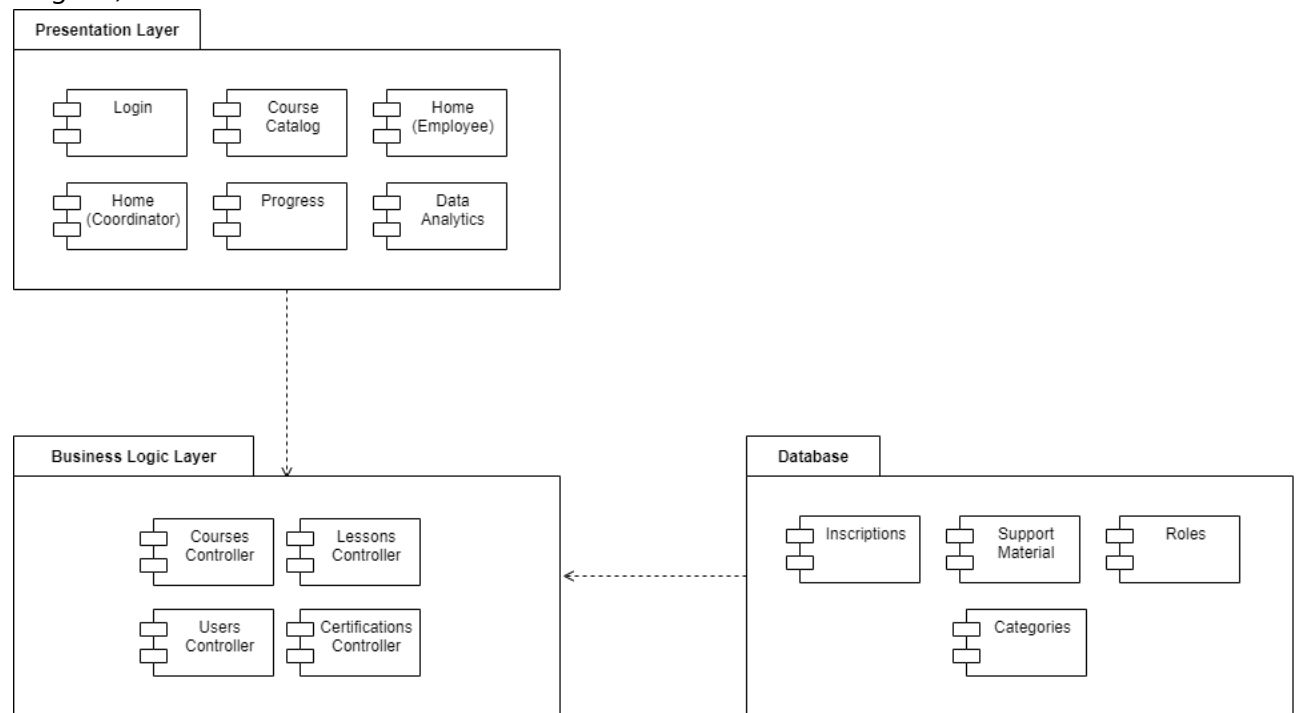


- A class diagram for the objects that will be instantiated, for the previously mentioned aspects (Include at least 5 elements of the diagram)

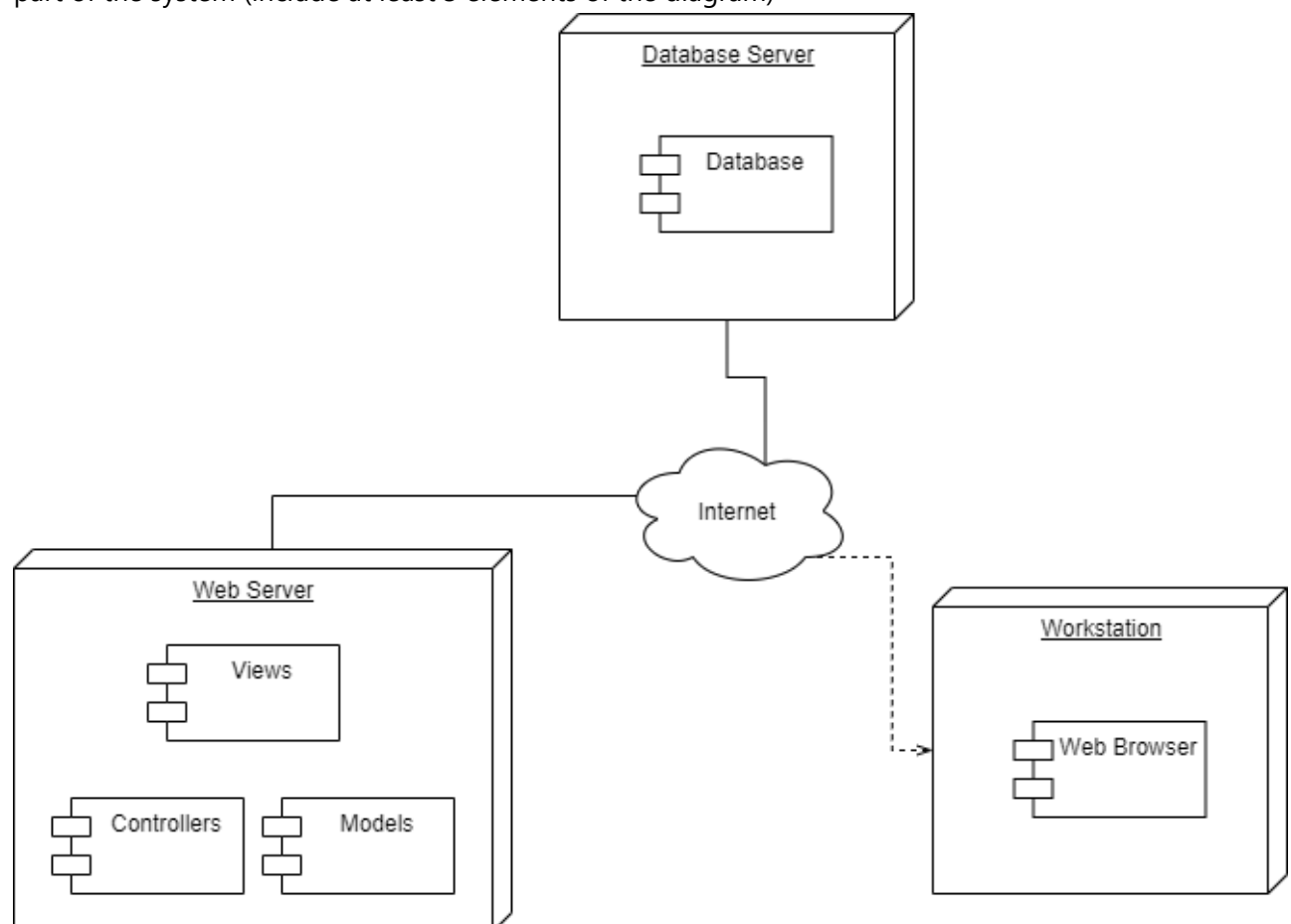


- A package diagram containing the component diagrams and the relationship between the database, user interfaces, controllers or intermediate layers - middleware (Include at least 3 elements of the

diagram)



- The distribution diagrams representing the system's physical structure, such as physical infrastructure, networks, storage and web servers, firewall, mobile devices or any other physical resource that will be part of the system (include at least 3 elements of the diagram)



### a) *Bañuelos Mendez Jordi's Conclusion*

This activity allowed us to gather the different requirements and components that will be part of the system and represent them graphically through diagrams, which are necessary to obtain a better idea of how the final product will work. Identifying each aspect is vital to guarantee a good functioning of the system, and having each one of these points represented along with their interactions through diagrams allows us to understand how the software should work.

### b) *Castillo Medina Edgar Antonio's Conclusion*

The platform we're going to develop is becoming a little clearer with the previous diagrams, because the requirements aren't just indications or requests anymore, now they have a representation that explains most of their functions and activities with each specific user of the system. I think that the general idea or environment for the platform is closer to get ready to begin its development.

### c) *Villanueva Romero Carlos Daladier's Conclusion*

After the realization of the different diagrams I was able to understand how important is the proper realization of them, since we must be quite clear and specific in the different elements that each diagram requires, in such way that it can be understandable for the development team, and also to prevent discrepancy between what was requested and what was developed by them.

### d) *Villegas Ramirez Luis Eduardo's Conclusion*

With this activity we have been able to recognize and define the roles and their interaction with the system, as well as the activities that each role will do, which is very useful to understand how the system should behave and prepare it for each specific case, defining rules and parameters.



## Rubric

Criteria	Description	Score
Instructions	Is each one of the points indicated in the instructions section fulfilled?	10
Development	Was each one of the points requested within the development of the activity answered?	60
Demostration	Is the student present during the explanation of the functionality of the activity?	20
Conclusions	Is a personal opinion of the activity included by each of the team members?	10



[Go to Home](#)