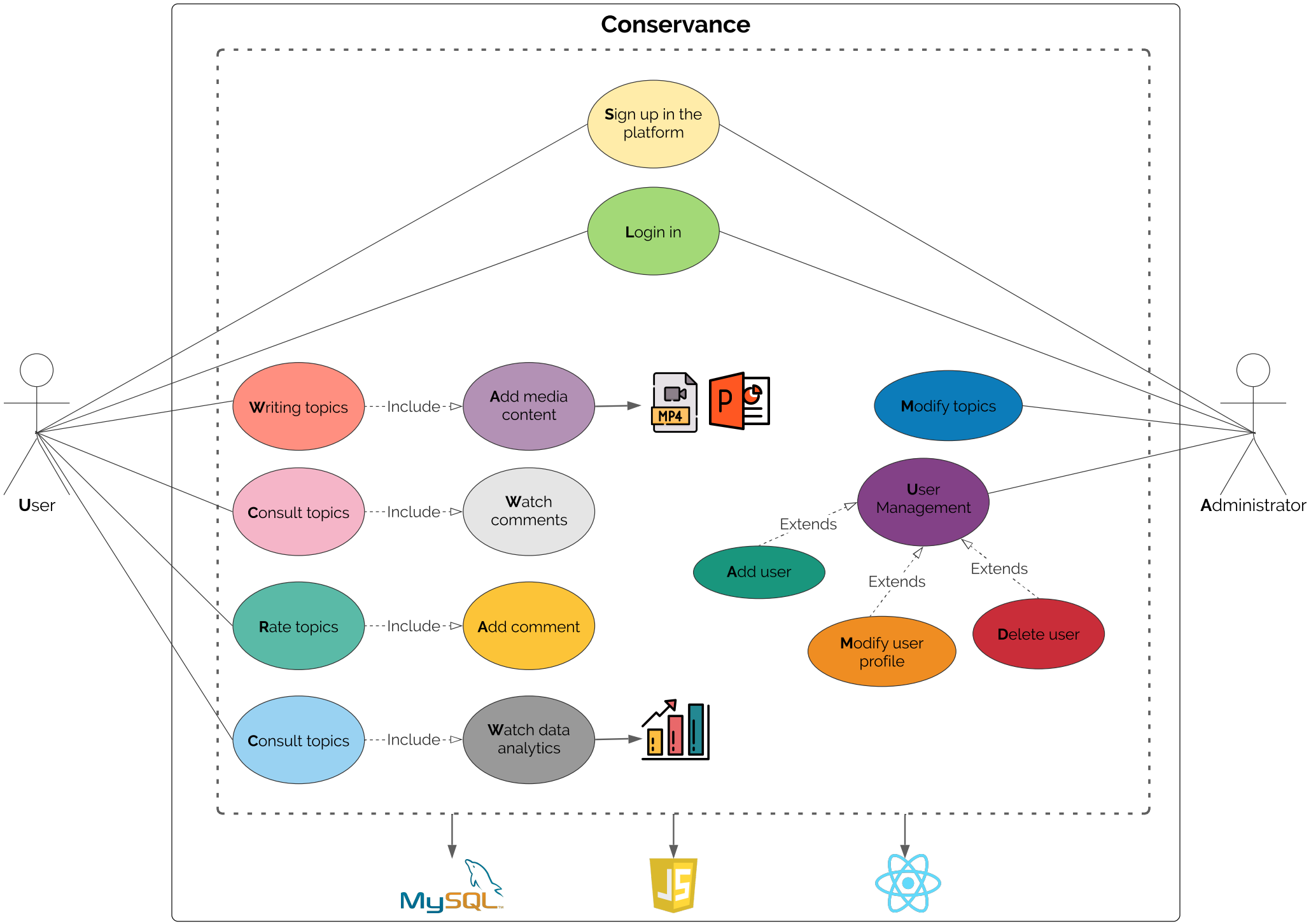
Introduction

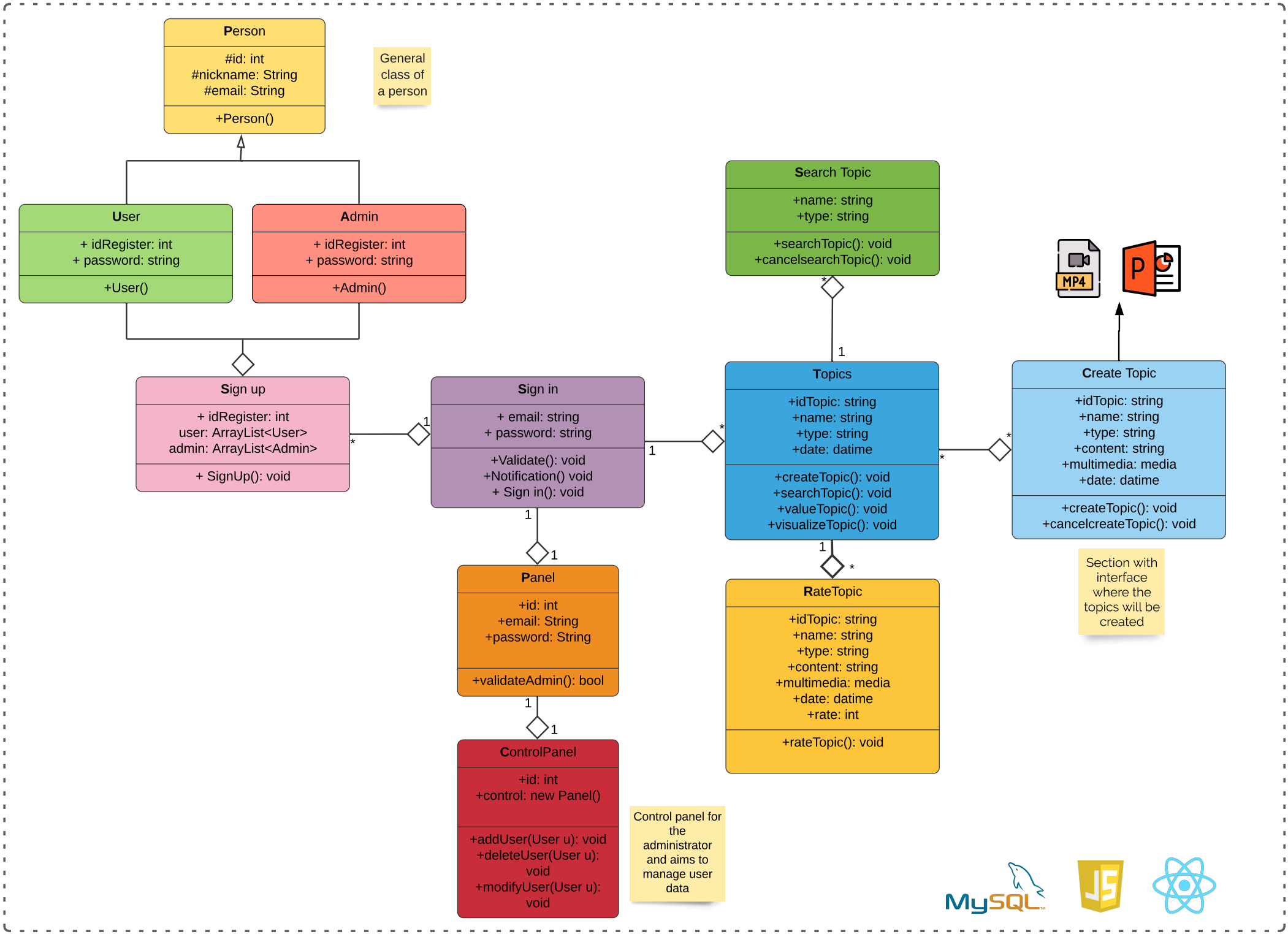
🏆 A2.2 Learning Activity

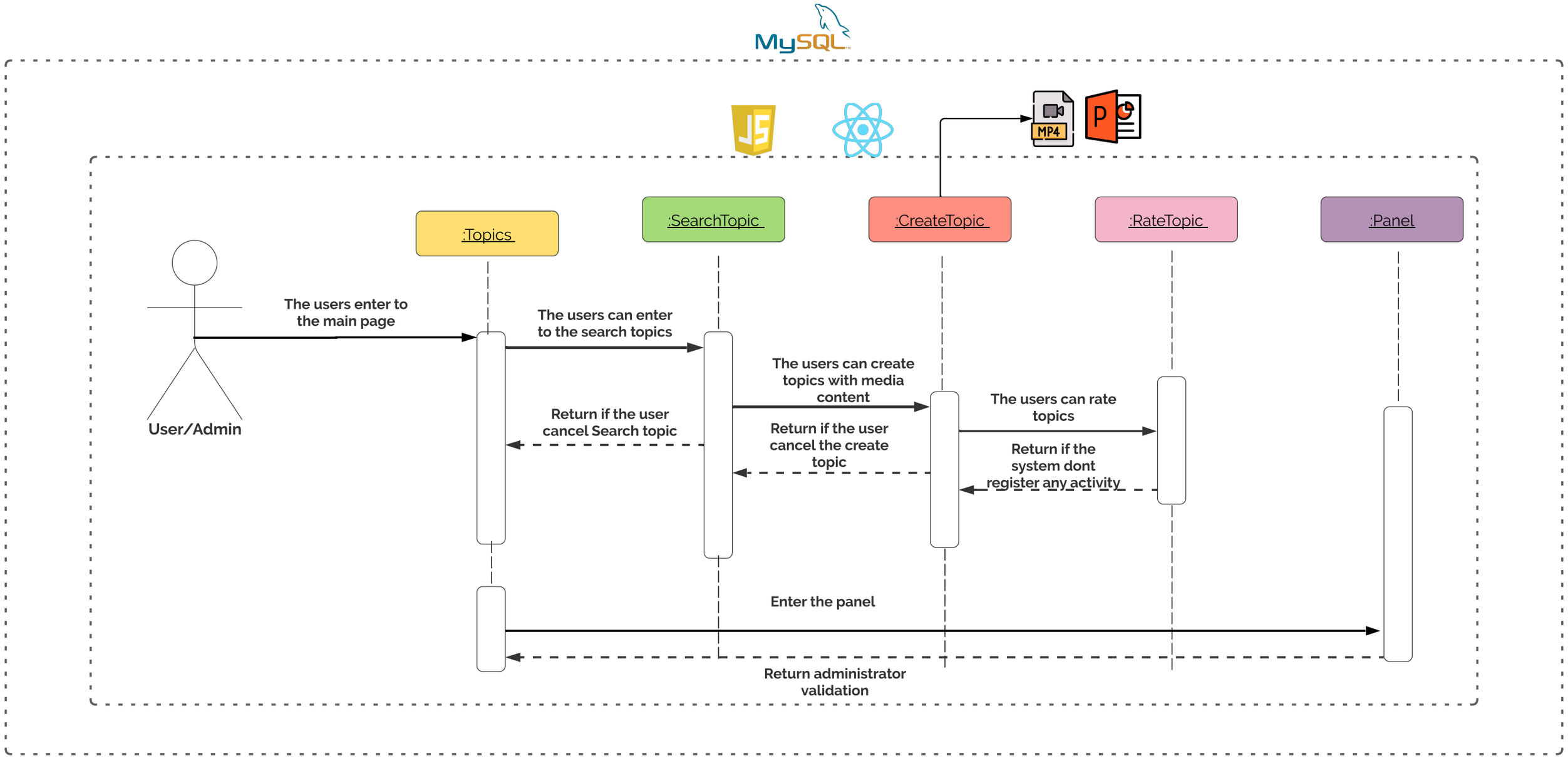
- Documentation of the system architecture based on the 4+1 model.

✏️ Development

1. Create diagrams for each of the views set in the 4+1 architecture model.
    - ☑ Scenario view: User case diagram.
    - ☑ Logical view: Class diagram.
    - ☑ Process view: Sequence diagram.
    - ☑ Developer view: Component diagram.
    - ☑ Physical view: Distribution diagram.
2. Each diagram should contain at least 3 elements within its representation.
    - ☑ Use cases (Include at least 5 diagram elements).
    - ☑ Sequence diagram (Include at least 5 diagram elements).
    - ☑ Class diagram (Include at least 5 diagram elements.)
    - ☑ Package diagram containing component diagrams (Include at least 3 diagram elements).
    - ☑ Distribution diagram (Include at least 3 diagram elements).
3. Indicate by annotations the own technologies that will have to be used, supported by images or illustrations that represent them.

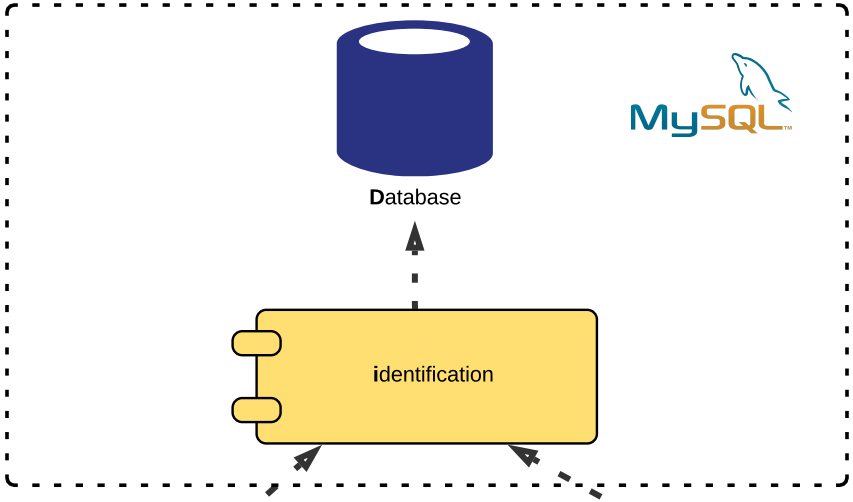

Conservance
Sign up in the platform
Login in
User
Writing topics
Include
Add media content
MP4
Consult topics
Include
Watch comments
Rate topics
Include
Add comment
Consult topics
Include
Watch data analytics
Modify topics
User Management
Extends
Add user
Extends
Extends
Modify user profile
Delete user
Administrator
MySQL

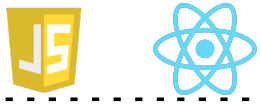

Person
#id: int
#nickname: String
#email: String
+Person()
General class of a person
User
+ idRegister: int
+ password: string
+User()
Admin
+ idRegister: int
+ password: string
+Admin()
Sign up
+ idRegister: int
user: ArrayList<User>
admin: ArrayList<Admin>
+ SignUp(): void
Sign in
+ email: string
+ password: string
+Validate(): void
+Notification() void
+ Sign in(): void
Search Topic
+name: string
+type: string
+searchTopic(): void
+cancelsearchTopic(): void
Topics
+idTopic: string
+name: string
+type: string
+date: datime
+createTopic(): void
+searchTopic(): void
+valueTopic(): void
+visualizeTopic(): void
Create Topic
+idTopic: string
+name: string
+type: string
+content: string
+multimedia: media
+date: datime
+createTopic(): void
+cancelcreateTopic(): void
MP4
Section with interface where the topics will be created
Panel
+id: int
+email: String
+password: String
+validateAdmin(): bool
RateTopic
+idTopic: string
+name: string
+type: string
+content: string
+multimedia: media
+date: datime
+rate: int
+rateTopic(): void
ControlPanel
+id: int
+control: new Panel()
+addUser(User u): void
+deleteUser(User u): void
+modifyUser(User u): void
Control panel for the administrator and aims to manage user data
MySQL

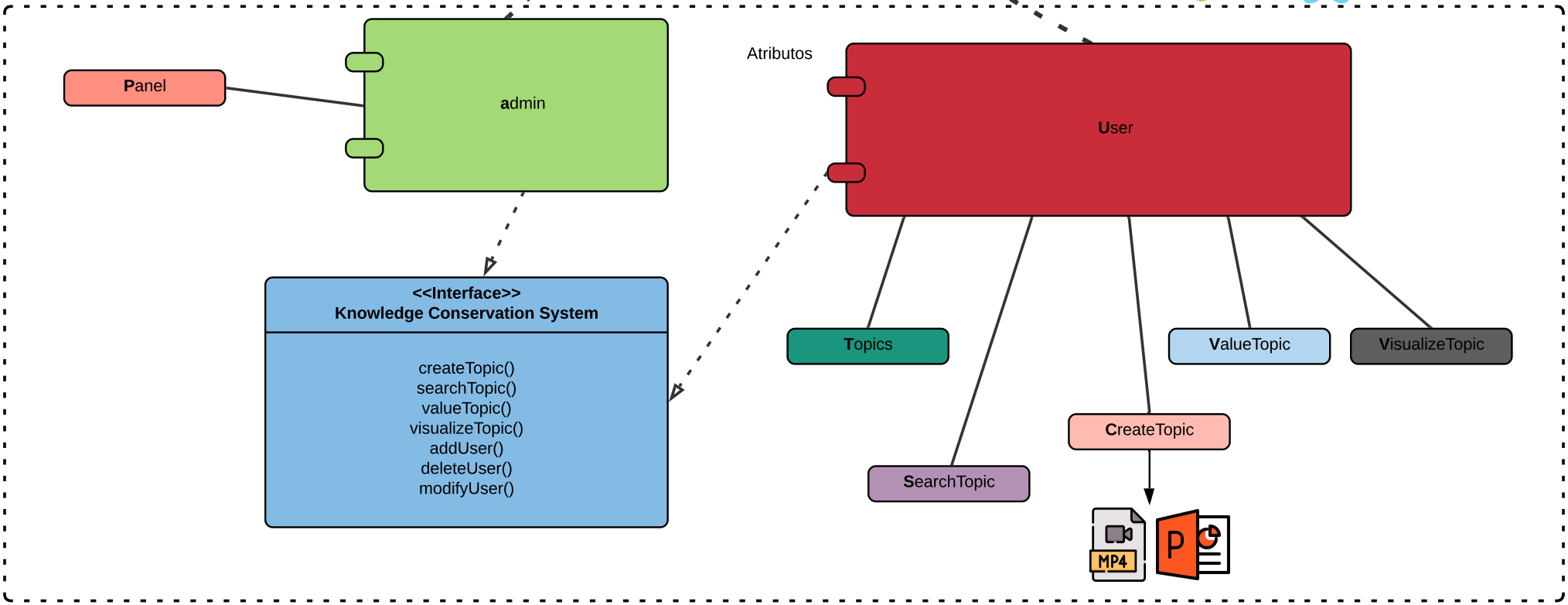

MySQL
MP4
:Topics
:SearchTopic
:CreateTopic
:RateTopic
:Panel
User/Admin
The users enter to the main page
The users can enter to the search topics
The users can create topics with media content
The users can rate topics
Return if the user cancel Search topic
Return if the user cancel the create topic
Return if the system dont register any activity
Enter the panel
Return administrator validation

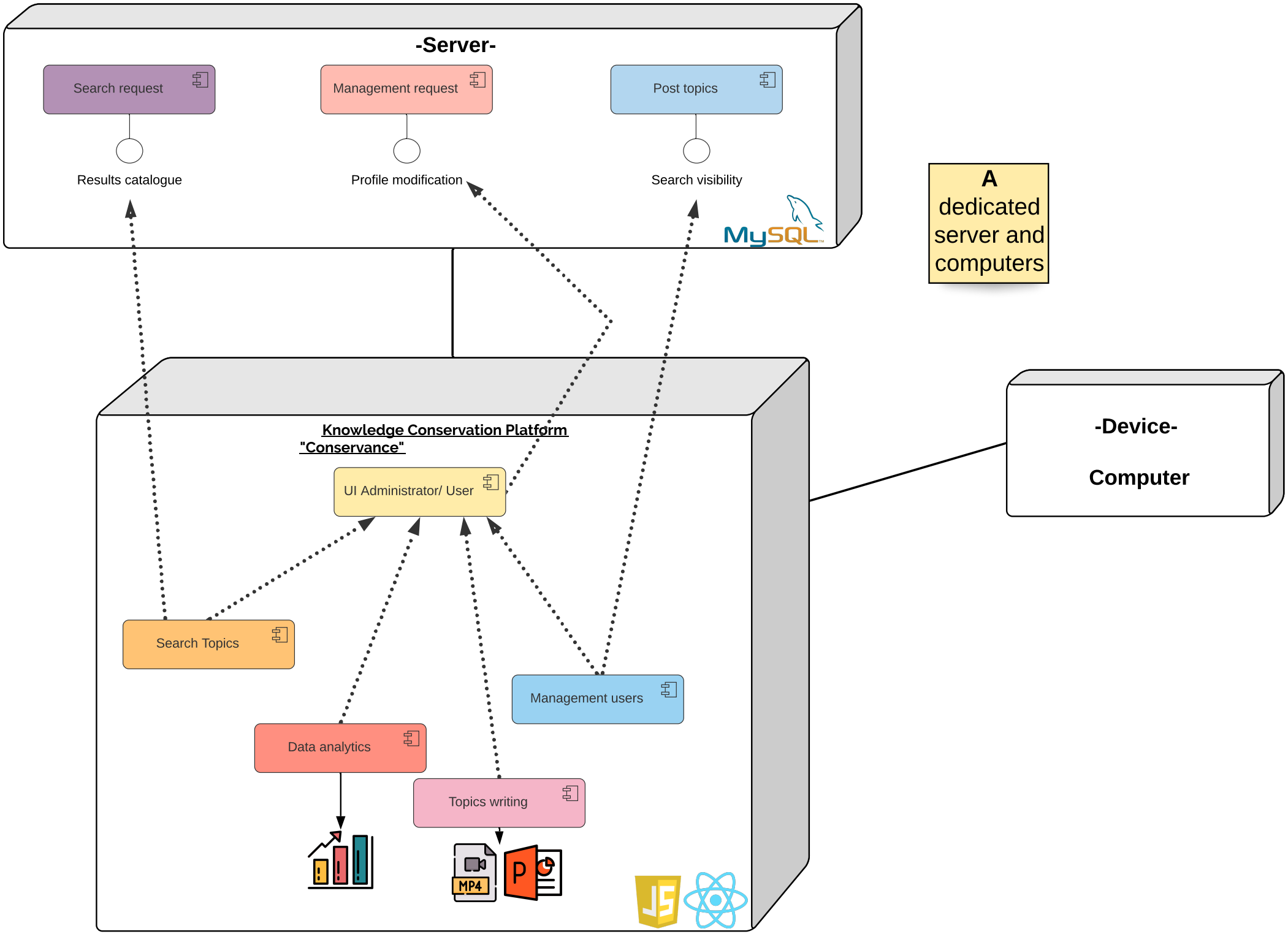
Conservance

Abstract methods in an interface or class and the data of the classes is stored in a database that validates access data








-Server-
Search request
Management request
Post topics
Results catalogue
Profile modification
Search visibility
MySQL
A dedicated server and computers
Knowledge Conservation Platform "Conservance"
UI Administrator/ User
-Device-
Computer
Search Topics
Management users
Data analytics
Topics writing
MP4

Conclution by Edson:

It was easy to implement it because we already had them from an activity that was previously done. The use of the 4+1 architecture is useful to organize the system through the functional and non-functional requirements that we already have. In addition, through the development of these diagrams it is possible to make annotations that will represent the technologies that are going to be implemented for the development of the project, in this case, the case study.

Conclution written by Antonio:

You can come to the conclusion of different views, in UML modeling and the views presented in the 4 + 1 architecture, have different elements, but in the end they have the same sense or are focused towards the same goal. With this assessment you can get to see with everything that is counted in the system that are in the development phase (documentation). It is important to review the requirements of the client that must be included in the system(functions that must have).it was very similar to me the approach of the diagrams where they pointed with different elements, but as I mentioned before they go towards gathering and capturing the same information.

Conclution written by Joseph

In conclusion, the diagrams that were made for this document were very similar to the previous ones created, only comments were added to make it much easier for the reader to identify what it is or the functionality. The 4 + 1 architecture gives us an approach to each view that has to be made so that both client and user can understand and once created it can move forward in the creation of said project. The functionalities that we show in each of the diagrams can fulfill much more extensive functions, but for this, a previous analysis of the weights for each structure was made.

Go to my GitHub repository