

# Introduction



## A2.3 Learning activity

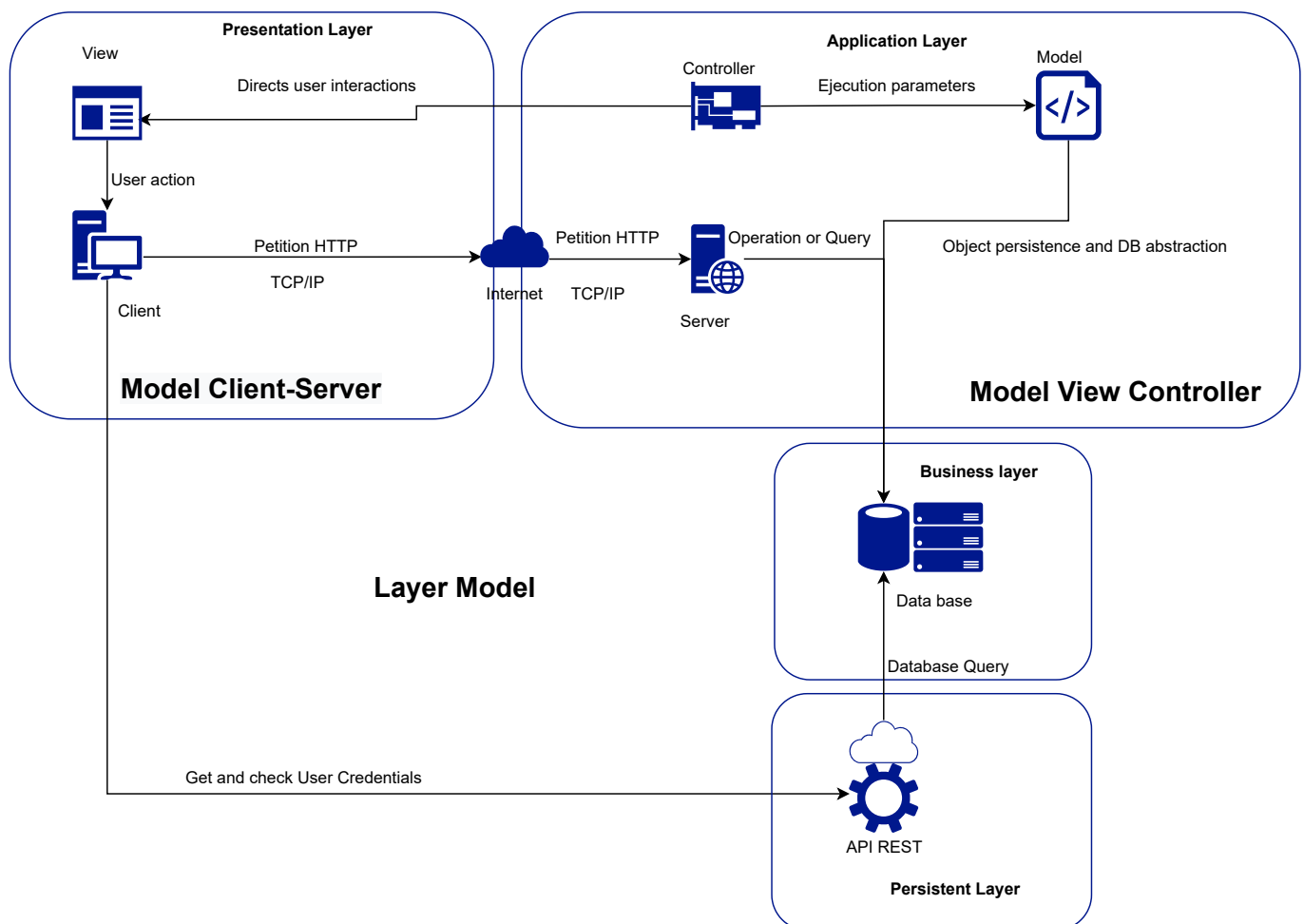
- Software architecture patterns








## Developing

1. Consider applying the following architecture patterns to the case study.

- ☐ Client-server architecture pattern
- ☐ Layered architecture pattern
- ☐ Architecture pattern Model view controller
- ☐ Pattern of architecture oriented to micro-services
- ☐ Architecture pattern Pipes and filters
- ☐ Event-based architecture pattern
- ☐ Repository or blackboard architecture pattern
- ☐ Publisher / subscriber architecture pattern
- ☐ Master-slave architecture pattern
- ☐ Point-to-point architecture pattern



2. Take as a basis the architectural views as well as the elaborated UML diagrams, to apply the pattern that you consider the most appropriate for each of the following scenarios.
  - 2.1 Considering that it is desired to **maintain and scale the system**, it is sought to develop the application through the decomposition of small independent and isolated services, which consume an external interface to communicate to a database server.  
 micro services
  - 2.2 Being the **confidentiality and security** attributes or requirements for the case study, it is sought to structure each of the components to be programmed in groups of sub-tasks, where each of these sub-tasks should communicate an intermediate layer and this to another upper layer.  
 layers
  - 2.3 Looking for the **availability** of the system, it is proposed to install two service servers, where these make their requests to a third server in which the database would be stored.  
 server/client
  - 2.4 Identifying that **data integrity** is a requirement, it is proposed to replicate and synchronize the database stored in the main server to another one, considering the main server as master and the secondary one as slave.  
 master/slave
  - 2.5 Considering that the client requested that each time a failure occurs, it must be provided alarms to the different users regardless of the place where they are, it is observed that You must hire a service provider to trigger the alarms and they reach users required.  
 publisher/subscriber

## Conclusions

### Carlos Gallardo

There is a great variety of patterns that allow us to work according to the specific problems that we are facing in our project. However, there are some that are or will be more common than the others (those found in the document). It is important to identify some architectural patterns that can help us, since having a pattern can work in a more structured and efficient way. The biggest challenge in the activity is to identify these patterns correctly as needed.

### Yessica Orihuela

In this activity about software architectures or patterns, one of the things that took a little more work was to know how to identify these patterns since many of them have similar elements. For that reason the point number 1 in which is an integration of different architectures was the most difficult and I consider that it is very important to know the different patterns since they serve as templates of software that is already proven to work in a certain way.

**Merari Cortes** In this activity I was able to see the software architecture I realized that there are many styles, designs and patterns of architecture, it was a bit difficult for me to identify and relate the activity, now I know that this activity is important for our project since we must know how to differentiate the patterns.

**María Sanabria** I found this activity quite easy since it's just a matter of identifying the software architecture. Once we have made every diagram the work is easier and things concerning the project are clearer than

before. The next step should be selecting the best coding patterns for the system and starting to implement the functions of the platform.

---

[Go to repository](#)