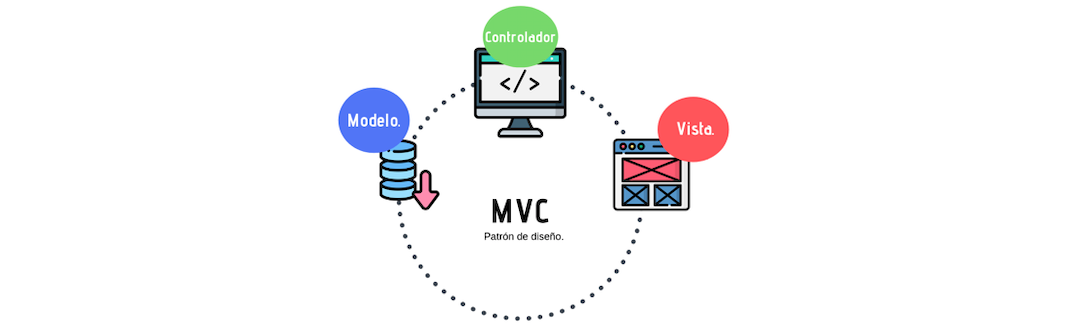
MVC structure,

The MVC pattern divides the application into three distinct parts: the Model, the View, and the Controller. Let's see in more detail what each of these parts consists of:

The Model: manages the application data. In our application, the model will be in charge of saving the information of all the bookmarks that the user has added. The model will not know anything about the view or the controller. Your task is to save and manage the information.

The View: represents the current state of the Model, without being in contact with it. The view is the "dumb" part of the application, its task is to display the information to the user.

The Controller: It is the link between the model and the view. It is in charge of "warning" the model when the user manipulates the view. In our application, the controller will be responsible for managing the changes that the user makes, such as adding or removing a bookmark.

* 

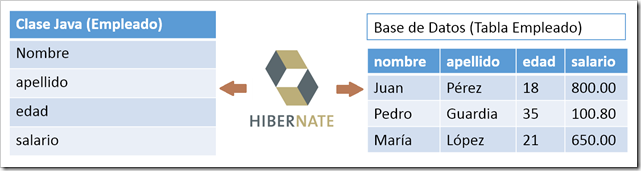
Using JPA with hibernate

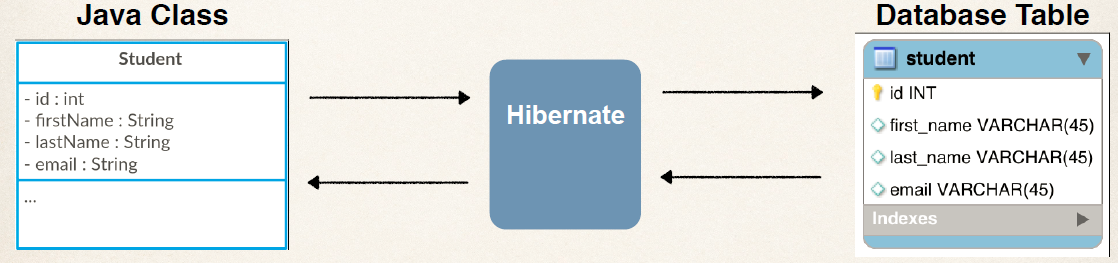
JPA:

Java Persistence API (JPA) is a framework that is part of Java, and offers a set of interfaces and APIs to solve the problem of storing objects in a relational database

HIBERNATE

Object-Relational mapping, O / RM, ORM, and O / R mapping) is a programming technique to convert data between the type system used in an object-oriented programming language and the use of a relational database as an engine persistence.





Dependency injection @Controller, @Entity etc.

The Dependency Injection pattern, also known as Inversion of Control, is a pattern that aims to achieve a more decoupled code, which will make things easier for us when conducting Tests and also allows us to change parts of the system more easily in case if necessary.

Requestmapping

The controller is in charge of redirecting or delivering information to the backend, it allows us to send structured information

Thymeleaf

Template engine for communication between the backend and the application controller

https://www.thymeleaf.org/documentation.html