Developers test

# Programming

I’ve choose **MySQL** data engine for this project, due besides of I’m in the habit to use that, it’s likely the most extended database engine and it’s a good choice for this kind of project of a dynamic web application. This is the database model to perform the basic case of booking a room:

BOOKING

**Idr (FK room.id)**

**Idc (FK customer.id)**

Datetime

Accepted

Payed

....

ROOM

**Id (PK)**

**Idr (IND idroom)**

**Idh (IND idroom)**

Floor

Door

...

HOTEL

**Id (PK)**

Name

Location

....

CUSTOMER

**Id (PK)**

Name

Surname

....

DATABASE MODEL

Regarding to the MVC framework, I can’t send you the code of that, due I have my own framework and it’s protected by author rights (me), due I spend a lot of hours to code it and I can’t send you if I’m not contracted to your company, but I can show you an simplified explain about how it works in the requested project of booking case:

VIEWER

* **Constructor**: Non required
* **Method called**: Return a string variable of returned html code depending the input arguments.

CONTROLLER

* Constructor: Initializes Viewer, Template and DataBase properties as objects to the main classes (by default, the loaded viewer match with the <CONTR>View-><ACTION>() request action. Also loads default template as /tpl/<CONTR>/<ACTION>.tpl file.
* **Method called**: Checks session data if required and high level permissions to use it. Optionally, calls the View class to get rendered data in html format to fill template variable as parts of code. Finally, returns the replaced template result as a single string (or throws an exception if error happens).

ROUTER

* Gets the url request. Detects the QS action parameter formed as <CONTR>/<ACTION> string
* Calls to the <CONTR>Controller-><ACTION>() class
* Plots the returned string in last calling method (or catch exception object to print an error).

MVC MODEL (BASIC)

Regarding to the view used, I use smarty template, due it’s very dynamical, it allows you to use iterators and conditionals and it’s very easy to implements and converts from the html template to the tpl format. Also, has a very active community and is very used in a lot of php projects around the world.

The views used to perform a booking are:

* View for list of available hotels of the customer selected zone and/or selected datetime.
* View for details of rooms in the hotel selected by the customer
* View for summary of selection room and additional services
* View for payment process
* View for final summary of the booking operation

The CRUD implementation is in php hardcoded language in GitHub.

## Web searches to do this part

* CRUD: Create, Read, Update and Delete, searched in Google and referenced to Wikipedia