

DEVELOPMENT OF WEB SOLUTION R.D.R

Julien JUNCKER¹

University of technologies, CASTELO BRANCO, Portugal,
`julien.juncker@viacesi.fr`

Abstract. In this document, we will talk about the webpages of R.D.R project. The goal of my work is to create Webpages, database and python script for the customer can see all information about our stock of object. He can watch, edit, add and remove object in the webpage. We also have an authentication system with user rights. If user with default right want to manipulate some data, a request will be sent to admin for confirm these modifications. The website is available in English and Portuguese language.

Keywords: Web solution, Webserver, Database, R.D.R website

1 Web architecture

To realize this solution, we need to have a specific architecture. We use some informatic technologies like database and Web server. I will explain you in detail these technologies.

1.1 Database

Database is an organized collection of data, generally stored and accessed electronically from a computer system. This organization permit to manipulate data more quickly and simply. We use a language called SQL to manipulate with them. In our case we use **MySQL** as database.

1.2 Webserver

Webserver allows the machine to analyze a user's requests (in http form), and return the file corresponding to the request.

In our case, we use **Apache** as Webserver

1.3 PHP – back-end language

The back end refers to parts of a computer application or a program's code that allow it to operate and that cannot be accessed by a user. Most data and operating syntax are stored and accessed in the back end of a computer system. Typically, the code is comprised of one or more programming languages. The back end is also called the data access layer of software or hardware and includes

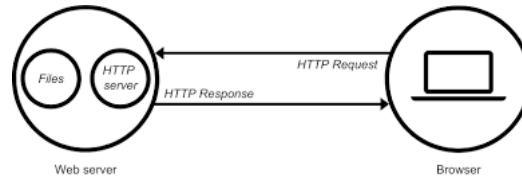


Fig. 1. Explanation of utility of Webserver

any functionality that needs to be accessed and navigated to by digital means. In our case, we use **PHP** as back-end language. PHP is a server-side scripting language. that is used to develop Static websites or Dynamic websites or Web applications. PHP stands for Hypertext Pre-processor, that earlier stood for Personal Home Pages. PHP scripts can only be interpreted on a server that has PHP installed. The client computers accessing the PHP scripts require a web browser only.

1.4 Front-end language

The layer above the back end is the front end and it includes all software or hardware that is part of a user interface. Human or digital users interact directly with various aspects of the front end of a program, including user-entered data, buttons, programs, websites and other features. Most of these features are designed by user experience (UX) professionals to be accessible, pleasant and easy to use. Generally, we use HTML, CSS and javascript in front-end.



Fig. 2. Explanation of composition of front-end

In our case, we use **HTML**, **CSS** with bootstrap and **Javascript** with JQuery. Bootstrap is an addon who give you better style for your website and JQuery is a library who simplify Javascript library.

2 R.D.R webpages

The goal is to create table with all object in database and manipulate it. We have some functionalities like:

- filtering table
- add object
- edit object

The web architecture is structured by MVC design pattern. If you want to know more about this pattern, you can show this part in next chapter. We have more details about this architecture and the code. In the webpage, you will see this page:

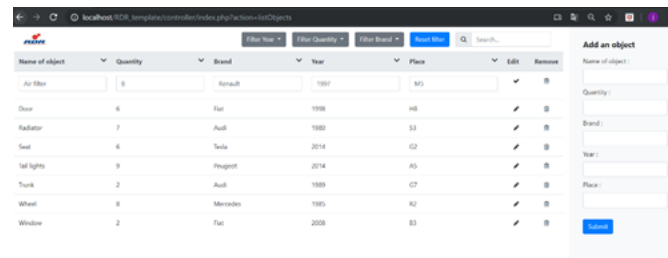


Fig. 3. RDR main page

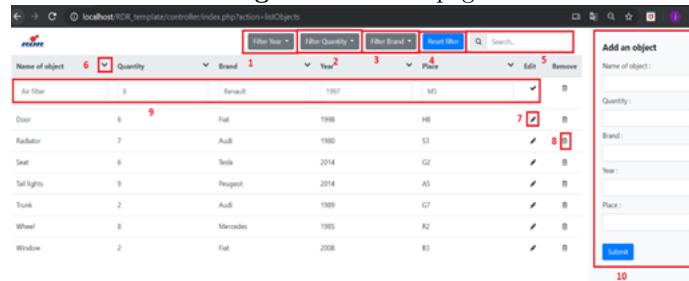
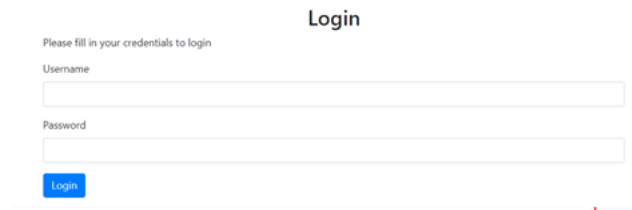


Fig. 4. RDR main page with functionalities

1. Filtering table by year with max and min value
2. Filtering table by quantity with max and min value
3. Filtering table by brand with dropdown selection
4. Reset Filter
5. Search bar to find object with specific research
6. Ordering each column by ascendant or descendant sort
7. Edit button click
8. Remove object into table and database

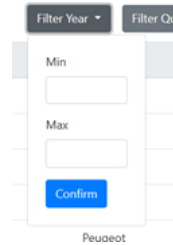
9. Editing object when you have clicked into editing button. To confirm, click on check button at same place of edit button
10. Add new object on table and database

If user isn't connected to website, it will be automatically redirect on user login page.



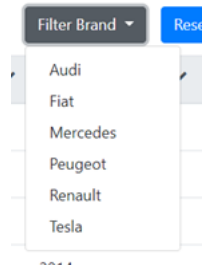
The image shows a login form titled "Login". Below the title, it says "Please fill in your credentials to login". There are two input fields: "Username" and "Password". Below the "Password" field is a blue "Login" button.

Fig. 5. User login page



The image shows a dropdown menu for "Filter Year". The dropdown is open, showing "Min" and "Max" input fields. Below these fields is a blue "Confirm" button. The word "Peugeot" is visible at the bottom of the dropdown menu.

Fig. 6. Min - Max dropdown interface



The image shows a dropdown menu for "Filter Brand". The dropdown is open, showing a list of car brands: Audi, Fiat, Mercedes, Peugeot, Renault, and Tesla. There is a blue "Reset" button next to the "Filter Brand" dropdown.

Fig. 7. brand dropdown interface

If you want to create new object but the name is same than other item in database, you will have a conflict.

You have three choice:

- Merge: you will merge the two objects and add quantity and place column
- Create new row: Create new object and keep the two elements
- Ignore: Ignore insertion of new object

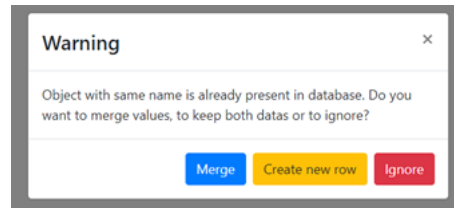


Fig. 8. Popup same object in database

With user rights, default client can't modify directly an object. Only administrator can do that. So, when user need to modify field, a request will be sent to administrator to confirm this operation. The client will see an alert when he wants to edit, add or remove object.

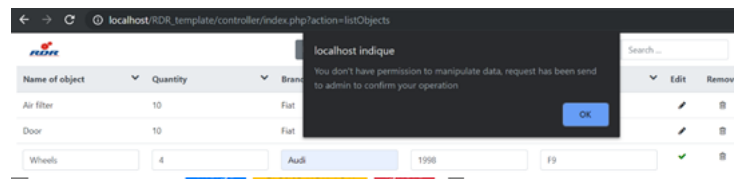


Fig. 9. Alert request sent to admin

In administrator panel, you'll see notification tab with number of modifications.

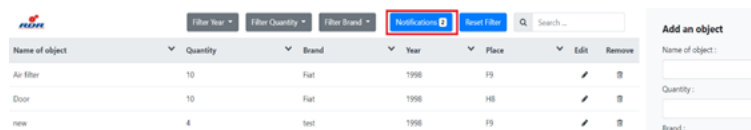
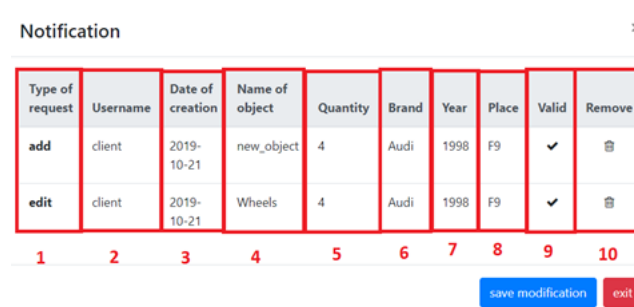


Fig. 10. Notification tab

In the popup you can confirm all operations. You'll see all request information.

1. Type of operation: add, edit, merge, remove
2. Name of user who send request
3. Date of creation of operation
4. Name of object to modify, add or remove
5. New quantity of object
6. New brand of object



Type of request	Username	Date of creation	Name of object	Quantity	Brand	Year	Place	Valid	Remove
add	client	2019-10-21	new_object	4	Audi	1998	F9	✓	
edit	client	2019-10-21	Wheels	4	Audi	1998	F9	✓	

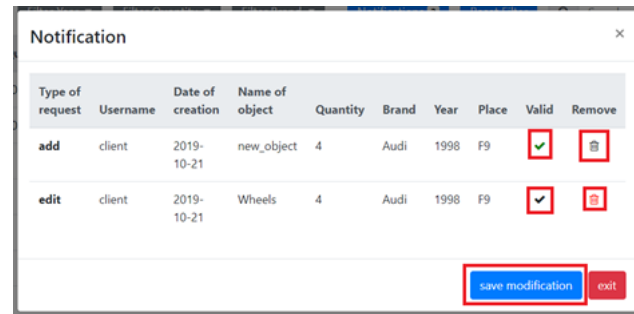
1 2 3 4 5 6 7 8 9 10

save modification exit

Fig. 11. Notification popup

- 7. New year of object
- 8. New place of object
- 9. Request validation button
- 10. Request remove button

To valid request, you need to click on valid or remove button of each request and to click on save modification.



Type of request	Username	Date of creation	Name of object	Quantity	Brand	Year	Place	Valid	Remove
add	client	2019-10-21	new_object	4	Audi	1998	F9	✓	
edit	client	2019-10-21	Wheels	4	Audi	1998	F9	✓	

save modification exit

Fig. 12. Notification popup confirm request

3 Web architecture explanation

As I said earlier, for the development of webpage, we use MVC design pattern. In this part I will explain you what it is and the architecture in R.D.R webpages.

3.1 What is design pattern

Design patterns are typical solutions to commonly occurring problems in software design. They are like pre-made blueprints that you can customize to solve a recurring design problem in your code.

You can't just find a pattern and copy it into your program, the way you can with off-the-shelf functions or libraries. The pattern is not a specific piece of code, but a general concept for solving a particular problem. You can follow the pattern details and implement a solution that suits the realities of your own program.

Patterns are often confused with algorithms, because both concepts describe typical solutions to some known problems. While an algorithm always defines a clear set of actions that can achieve some goal, a pattern is a more high-level description of a solution. The code of the same pattern applied to two different programs may be different.

An analogy to an algorithm is a cooking recipe: both have clear steps to achieve a goal. On the other hand, a pattern is more like a blueprint: you can see what the result and its features are, but the exact order of implementation is up to you.

3.2 MVC design pattern in PHP

PHP MVC is an application design pattern that separates the application data and business logic (model) from the presentation (view). MVC stands for Model, View and Controller.

The controller mediates between the models and views.

Think of the MVC design pattern as a car and the driver.

The car has the windscreens (view) which the driver (controller) uses to monitor traffic ahead then speed or brake (model) depending on what he sees ahead.

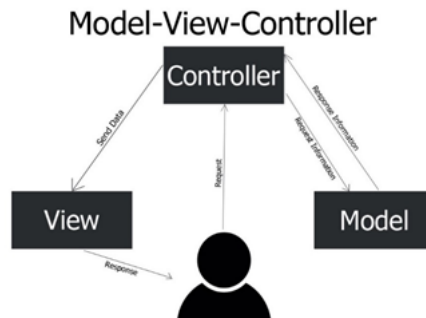


Fig. 13. MVC design pattern

Why use PHP MVC Framework? .

PHP MVC Frameworks simplify working with complex technologies by:

- Hiding all the complex implementation details

- Providing standard methods that we can use to build our applications
- Increased developer productivity, this is because the base implementation of activities such as connecting to the database, sanitizing user input etc. are already partially implemented
- Adherence to professional coding standards

Composition of MVC .

Model: This part is concerned with the business logic and the application data. It can be used to perform data validations, process data and store it. The data can come from:

- Flat file
- Database
- XML document
- Other valid data sources

Controller: This is the part deals with the users' requests for resources from the server. As an example, when the user's requests for the URL

Corollary 1. *.../index.php?object=list*

, the controller will load the products model to retrieve the products data then output the results in the list view. In a nutshell, the controller links the models and views together depending on the requested resources.

Views: This part deals with presenting the data to the user. This is usually in form of HTML pages.

3.3 R.D.R architecture

The architecture is made in this way:

- Config: Contains DB connection and configuration and URL path of the webproject
- Controller: Controller part of the project
- View: View part of the project
- Model part of the project
- Image: Contains all images present in the website

I will explain further the MVC part. First, we will speak about Controller.

Controller .

As explain earlier, he mediates between the models and views. In our case, He will receive HTTP POST request by the view and call the model to execute a specific action. In R.D.R, for the table of all objects in the dashboard, we need to receive data in database.

Controller part is composed of multiple files:

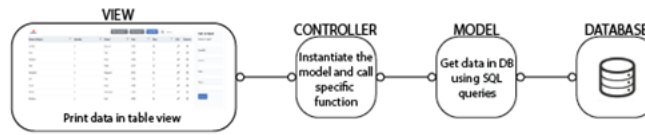


Fig. 14. Explanation of MVC design pattern in RDR project

- Index.php: It is used that hub, he receives all the http queries by the view and choose specific action.
- ObjectController.php: He contains all the specific action you can do with object of cars. For example, if you need to receive pieces of cars to print on view, you will develop this functionality of objectController part.
- UserController.php: He contains all the specific action you can do with user. For example, if you need to login on page, you will develop this functionality of userController part.

When Index.php receive http request. This request has a specific structure. When view part send request, the structure is this:

Corollary 2. *<http://website.com/index.php?action=listObjects&name=Wheels>*

In this url, first we have the protocol http: HTTP:// Secondary, we have the URL of index.php: WEBSITE.COM/INDEX.PHP Finally, you can add more custom action in this url:

Corollary 3. *ACTION=LISTOBJECTS&NAME=WHEELS*

When index file will receive the request, he will be going to split the action part (action="") to know what he need to do.

After that he instantiate the specific controller and call function with some parameters. for example, in the query that I introduced you, the query means to call "LISTOBJECTS" function with the parameter "WHEELS".

The query permit to receive all the information of wheel object on the database. The function to call the model is in ObjectController file because this action is linked at "OBJECT" comportment.

Model .

In R.D.R project, the Model is used to get or set data in database. After the instantiation of specific model class and the call of function, the model needs to execute the order that asked him. In R.D.R, we have four files in the model:

- ObjectModel.php: Execute object queries and return response to controller
- Object.php: Class contains all attributes of Object item
- View: View part of the project
- UserModel.php: Execute user queries and return response to controller

- User.php: Class contains all attributes of User item

Object class is very special. The only goal of this class is to store all the attributes of object, like this name, this quantity, this year ... In the secure way to manipulate object values. It is the same functionality with User class but with user information like username, password, right