

## General description

This practice is to fix the architecture of the final project. You have to define views that will be available, the structure of objects and the structure of the database (tables and fields). delivery of this practice must be a memory describing the Structure1, and a prototype functional with two features of the final project. memory doesn't have a format fixed, but must follow the structure indicated in the following sections.

## List of scripts for views

In the previous practice you added sketches of the different screens are viewing the different users, but not so exhaustive. in this practice have to list all scripts that will be required for all views. you can build a script for each screen, or the same script can be responsible for generating several screens. in this chapter of memory should be detail all scripts views to be implemented to give support to the interaction of the Web. for each you script have to indicate briefly what will serve. where appropriate, you can add diagrams indicating the flow of the application of a script to another. list of scripts additional as we have seen in class, an application can have scripts additional containing definitions of classes, the logic of the application or abstraigan management database. in this chapter of memory should be detail all scripts additional (not match views of the application) including the project.

## Structure of the database

In this chapter you to detail the structure of the database, indicating tables, fields and relationships.

For each table must indicate your purpose, detail scripts accessing the table (indicating if only consult or also modify) and explain the fields contents.

Explanations can be accompanied by diagrams in formats suitable to represent the structure (for example, but not necessarily diagrams of entity-relationship). P

prototype functional project based on the documentation generated in the previous sections, you must implement completely (including access to databasa):

- the login of the application

- at least 1 full functionality of the application (other than registry, and to agree with professor). in principle format is free, but it must comply with a series of mandatory requirements. style code tags, format and styles whole page must be written using documents html5 valid, without attributes or deleted items or not recommended. as a reference, can be used notes of the subject 2 or specification formal w3c. do not use any type of label or attribute format (e.g. <B>, <I>, <Center>, align, border, cellpadding, etc.).

all styles should be applied using CSS.

it will value especially proper use of CSS, especially model of boxes and advanced positioning. I will value specially cleaning HTML code and PHP (indentation, comments, structure uniform, orientation object, etc.), and the correct organisation according to pattern MVC of the files that make up the website.

## images

all pictures the Web must be contained in a folder IMG, located at the same level that the files main. this folder is due to deliver also, and all pictures must be in PNG format or JPG.

## Architecture

architecture of the application is assessed through architectural reasoning and understandable system. in particular, we will value that distinguishes between PHP scripts dedicated to generate views and scripts support (definitions class libraries support functions, abstraction of the database, etc.).

PHP code or JavaScript you can add anything on JavaScript. if you want to use libraries or frameworks open source, use must be justified and agreed with professor.

publication the Web usually the Web not run in local, but are stored in a server.

mandatory the publication of the page in virtual machines in laboratory.

in addition to delivery through the virtual campus, also need to leave configured the application + the database for testing in VMS laboratory. also must provide a user with administrator access to the application.

**Delivery**

Practice should be delivered before the 23:55 Wednesday April 27, through the virtual campus. a single .zip file that must contain a document in PDF format with memory as is has been described in the previous sections. a file readme.txt to indicate what functionality of the application is implemented completely in this delivery. A .zip with the prototype functional of the application, including the export of the database. enough with that give one of the members of the team. the name archive must be p2-nombreproyecto.zip.