**Students Database**

Technical Design

**Document Revision History**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ver. | Date | Author | Reviewed by | Approved by | Changes |
| 0.1 | 01/01/2011 | Someone | Someone else | Someone else | Initial draft |
|  |  |  |  |  |  |

Table of Contents

[1 Title 1 3](#_Toc307307689)

[1.1 Title 2 3](#_Toc307307690)

[1.1.1 Title 3 3](#_Toc307307691)

# General Design

## Objectives

The two main objectives are:

1. Good separation of responsibilities into small components, to ease the development by several people, and making easier to build the application step by step.
2. Keep the design simple, so anyone can easily be part of the development, without long training

To reach the first objective, each component will:

* be located into a separate directory
* have the responsibility of his own tables and data in the database, and provide functionalities to access for other components
* defines its own web pages, if applicable

To reach the second objective:

* we will not use a complete framework which may be long to learn, but may use small part of it or small libraries at some points to avoid doing something that already exist (like importing Excel files…)
* but still we will implement few principles usually provided by frameworks, as good design practices

## Structure

In order to ensure security, and enforce every component follow the same good practices, the components will not be directly accessible. Instead, only one entry point will be provided (index.php), and every request will be processed by this entry point. This entry point will then analyze the request, and then, if it is correct, it will dispatch it to the right component. However this feature will be hidden from the user point of view.

Each component, located in its own directory (component/<component\_name>) will follow the structure below:

* page/<page\_name>.php
* static/<filename>
* service/<service\_name>.php
* <component\_name>.inc

The *page* directory contains one file for each screen provided by this component. This file should have the extension *.php*, all other files in this directory must not have this extension.

The *static* directory will contain static resources (images, scripts…) which is not dependent of the user using the application, and thus can be easily cached by web browsers or web proxies, even among different user sessions.

The *service* directory will contains one file for each service provided by this component. A service is different from a page, in the sense that a page is in HTML format to be display in a browser, while a service may be return data in any format, is not intended to be displayed, and may be called either by a screen, or by a third-party system.

The file <component\_name>.inc will:

* Ensure every module implements security:
  + Declare the list of pages, together with the needed access rights to access the page
  + Declare the list of services, together with the needed access rights to access the service
* Provide any other functionalities that may be accessed by other modules

From outside, the files will be accessed using the URLs as follow:

|  |  |  |
| --- | --- | --- |
| Type | URL | Will access in the structure to |
| Page | /dynamic/<component>/page/<page> | /component/<component>/page/<page>.php |
| Service | /dynamic/<component>/service/<service> | /component/<component>/service/<service>.php |
| Resource | /static/<component>/<filename> | /component/<component>/static/<filename> |

The reason to separate dynamic elements from static elements into two different sub-directories from outside point of view is technical: to allow caching (no cookies outside of /dynamic/)