FMNSWEBDOC – WEB BASED SYSTEM FOR DOCUMENTATION

A. Stoilov and V. Hristov

South-West University “Neofit Rilski”
Blagoevgrad, Bulgaria
Ivan Mihailov str. 66
E-mail: antonstoilov@abv.bg

Abstract

The system for documents exchange in one faculty is one of the most important part of program support of them. The purpose of paper is to present one possibility for using PHP, MySQL and Apache for realization of this system. The present web based system is applicable for different faculty. The real application of this product is present and running in web. The security system was developed.

1. Introduction

The FMNSwebDoc is an attempt to make web based systems for document storage. The instruments used for this realization are PHP and MySQL. This system is orientated in UNIX based servers such as Apache.

Documents are the basic building blocks of FMNSwebDoc. The software has possibility of doing the instance of documents from basic templates. All processes are done in Internet.

Other very important parts are the users. Each user has their own login name and password. Users can only alter items that they own, or that they administer. This means that individual users cannot make changes to other users' items. There are two special types of users:

- Users can be made into administrators (also known as 'superusers'). This overrides the normal access controls; an administrator can alter or edit anything.
- Restrict users are the persons how use this software only in his work.

Most projects or tasks have a group of users working together in one specific area. A usergroup is a group of users who share a similar work area. Notification emails can be sent to the usergroup, rather than just an individual user. This is available now and usergroups respond to the departments in faculty.

FMNSwebDoc has been developed on Red Hat Linux 9.0, Fedora’2006 and Debian (stable/testing) and run in a production environment with initially, PostgreSQL, and later MySQL databases. FMNSwebDoc will only run on PHP 4.2.0, or higher [1-4]. Earlier versions of PHP do not support $_POST, $_GET or $_REQUEST global arrays, which are necessary for FMNSwebDoc to function. The setup program of the system check for early versions of PHP and will record an error.

2. Basic conditions

With a web server such as Apache and MySQL, you have most of what you need to develop a web database application. The key glue you need is a way for the web server to talk to the database; in other words, a way to incorporate database operations into web pages. The most popular glue that accomplishes this task is PHP.

PHP is an open source project of the Apache Software Foundation and it's the most popular Apache web server add-on module, with around 53% of the Apache HTTP servers having PHP capabilities [1-3]. PHP is particularly suited to web database applications because of its integration tools for the Web and database environments. In particular, the flexibility of embedding scripts in HTML pages permits easy integration of HTML presentation and code. The database tier integration support is also excellent, with more than 15 libraries available to interact with almost all popular database servers. Apache, MySQL, and PHP can run on a wide variety of operating systems.

PHP is a simple yet powerful language designed for creating HTML content. PHP runs on all major operating systems, from Unix variants including Linux, FreeBSD, and Solaris to such diverse platforms as Windows and Mac OS X. It can be used with all leading web servers, including Apache, Microsoft IIS, and the Netscape/iPlanet servers.

The language is very flexible. For example, you aren't limited to outputting just HTML or other text files – any document format can be generated. PHP has built-in support for generating PDF files, GIF, JPG, and PNG images, and Flash movies.

One of PHP's most significant features is its wide-ranging support for databases. PHP supports all major databases (including MySQL, PostgreSQL, Oracle, Sybase, and ODBC-compliant databases), and even many obscure ones. With PHP, creating web pages with dynamic content from a database is remarkably simple.

Finally, PHP provides a library of PHP code to perform common tasks, such as database abstraction, error handling, and so on, with the PHP Extension and Application Repository (PEAR). PEAR is a framework and distribution system for reusable PHP components.

The conception of development of FMNSwebDoc is present in fig.1. There is present three tiers corresponding with three level of communication. The first level is communication between client and server. The requests are available by Internet. Next level of transfer of requests is reports from database.

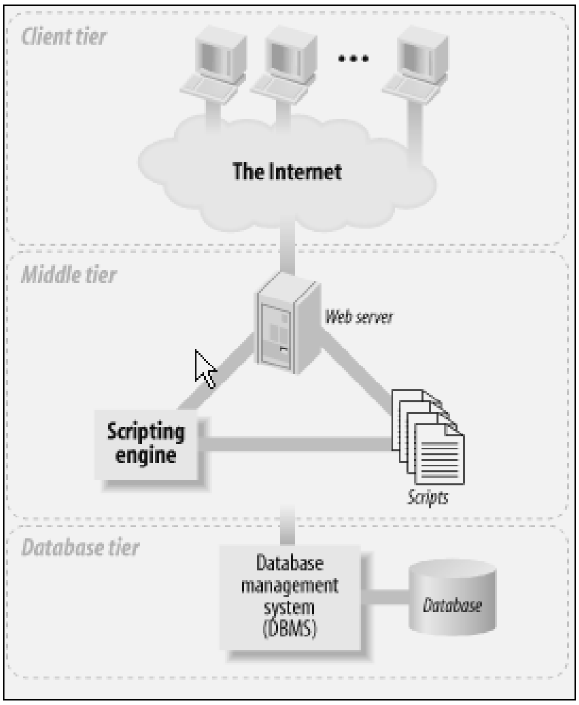


Fig. 1. Schematic of transfer of requests in FMNSwebDoc

The main advantages of this structure are: high security level, speed, separate of task and optimization of requisite hardware devices.

Most of the services we enjoy on the Web are provided by web database applications. Web-based email, online shopping, forums and bulletin boards, corporate web sites, and sports and news portals are all database-driven. To build a modern web site, you need to develop a database application.

The original FMNSwebDoc code was specifically written for a PostgreSQL database. The code now has a database abstraction layer and runs either MySQL or PostgreSQL. A calendar section and setup programs have been added. Documentation for the program has been written. Work has also been done to enable FMNSwebDoc to work with Unicode (UTF-8) and other multibyte encoding systems. This is offered as an optional alternative, since MySQL prior to version 4.1 and PHP prior to version 4.3.0 have poor multibyte support.

The database types in the config file are:

- Postgresql => PostgreSQL abstraction layer.
- Mysql => standard MySQL abstraction layer
- Mysql_innodb => MySQL abstraction layer with innodb transaction support

In setup the administrator have possibility to choice on of them.

PostgreSQL is an Object-Relational Database Management System (ORDBMS) that has been developed in various forms since 1977. It began as a project named Ingres at the University of California at Berkeley. Ingres itself was later developed commercially by Relational Technologies/Ingres Corporation. In 1986 another team led by Michael Stonebraker from Berkeley continued the development of the Ingres code to create an object-relational database system called Postgres. In 1996, due to a new open source effort and the enhanced functionality of the software, Postgres was renamed to PostgreSQL, after a brief stint as Postgres95 [4]. The PostgreSQL project is still under very active development worldwide from a team of open source developers and contributors. PostgreSQL is an open source project. Open source by definition means that you can obtain the source code, use the program, and modify it freely without the confines of proprietary software. In the database world, open source means that you have honest access to benchmarking numbers and performance statistics, which companies such as Oracle prohibit. Open source, also means, that you are free to modify PostgreSQL to fit your particular needs.

The database used transactions. The PostgreSQL abstraction layer always uses transactions because:

- They provide better data security in the event of power failure or hardware/software crashes during database write operations. Incomplete data is not witten to the database, and data corruption or loss is avoided.
- Without transactions it is possible for two users to alter the same data simultaneously, leading to data corruption. With transactions this is prevented by design.

The mysql_innodb abstraction layer also uses transactions. For MySQL version 4.0, and above, innodb database support is enabled by default. For these versions (and configured 3.23.xx versions) it is recommended that the database be created with the mysql_innodb database creation script, and the mysql_innodb abstraction layer is used. For MySQL version 3.23.xx the database server must first be correctly configured to allow innodb databases to be used.

3. The web based system FMNSwebDoc

The scheme of the necessary tables in database is show in fig. 4. These tables are creating in case of choose MySql database. All tables are 14.

After creating of these tables is execute this two rows for initialization of "users" and "config" tables:

INSERT INTO users (id, name, fullname, password, email, admin, deleted)

VALUES (1, 'admin', 'Administrator', '0192023a7bbd73250516f069df18b500' , 'please_edit@my_domain.com', 't', 'f');

INSERT INTO config (globalaccess, groupaccess, project_order, task_order) VALUES ('checked', '', 'ORDER BY name', 'ORDER BY name');

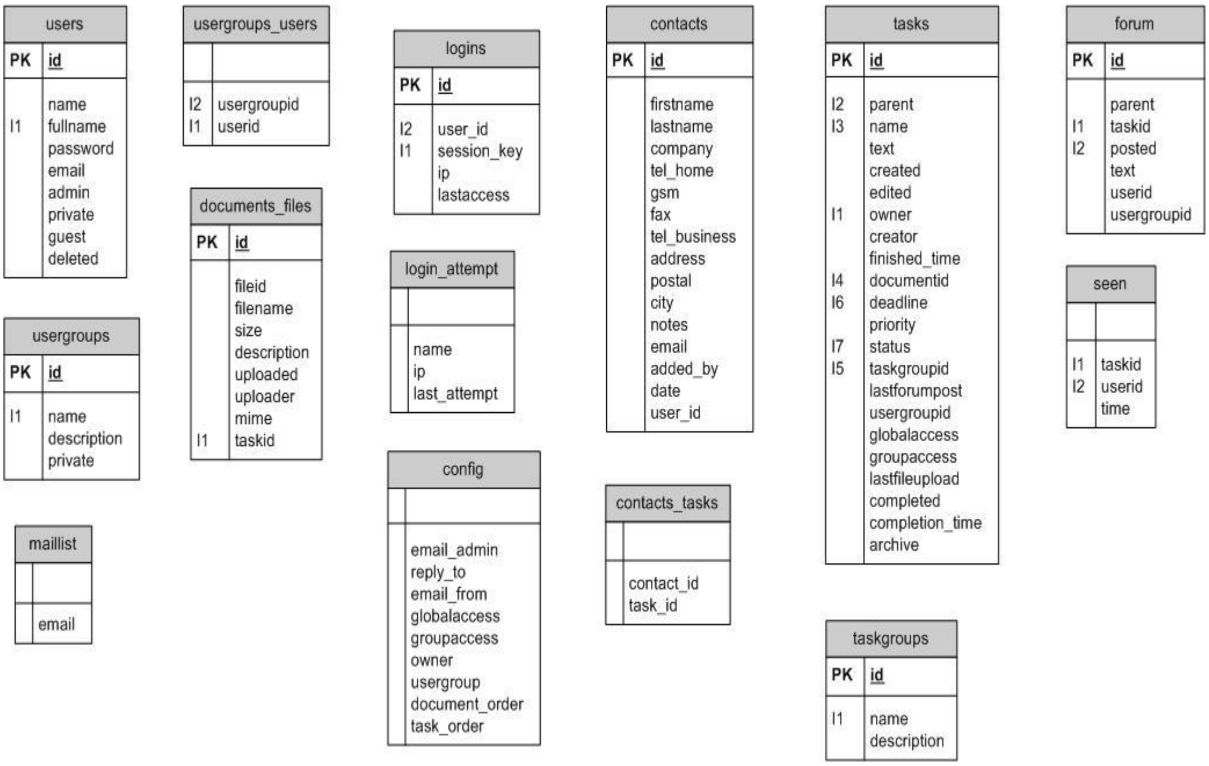


Fig. 2. Scheme of tables in database

Actualy, when finished the setup procedure we have one "super user": "admin" with password "admin123". With this account we create other accounts for each department. This data is remembered in "user" table.

For every department is created individual "usergroups", the data are in the same name table. We have and interstitial table – "usergroups_users". This table includes information for correspondence between user and usergroups, and for rapid recognition of usergroup of some user.

Every document generated by system, must by uploaded on the server. This procedure is manual, because the number of generated from templates documents in real is much more them the necessary and if all of these documents are uploaded automatically, the stability of system will be in danger. The information about these files is in "documents_files" table. The procedure for upload is shown in fig. 3.

Other very important tables are:

- "config" table – the table with value for main variables. This is initial values when system started
- "contacts" table – information for people in this work area.

We have and the table for forum. The form is place where every user have possibility to discussed with other users a theme how is important for all department.

When the program is started on the monitor is display the login. There is doing the check for user and privileges. In FMNSwebDoc are recognized two main accounts: administrator and user.

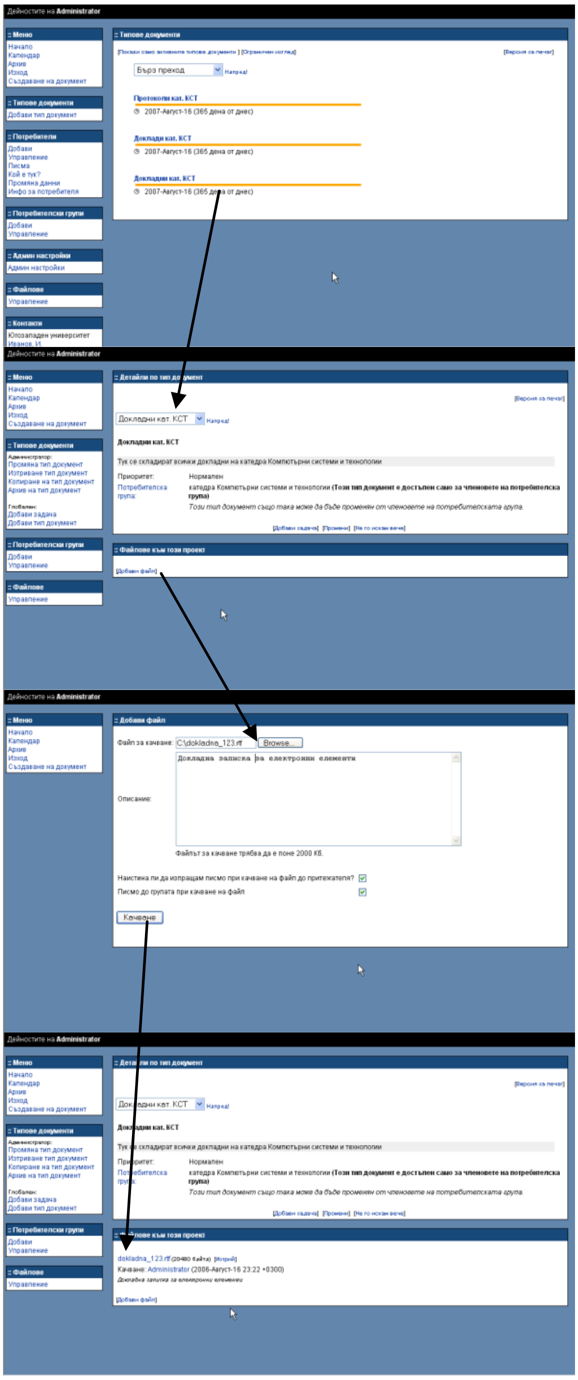
This process covered main aim of this system, namely creation of a catalogue for uploaded documents in one department.

Tasks are another part of documentation process in faculty. For this are create three tables:

- "tasks"
- "taskgroups"
- "contacts_tasks"

The names of tables are completing eloquent and the fields of these tables are present in fig. 2.

For assure the login was created two tables in database "login" and "login_attempt".



Югозападен университет
Иванов, И.

:: Типове документи

[Покажи само активните типове документи] [Ограничен изглед] [Версия за печат]

Бърз преход Напред!

Протоколи кат. КСТ
2007-Август-16 (365 дена от днес)

Доклади кат. КСТ
2007-Август-16 (365 дена от днес)

Докладни кат. КСТ
2007-Август-16 (365 дена от днес)

:: Детайли по тип документ

[Версия за печат]

Докладни кат. КСТ Напред!

Докладни кат. КСТ

Тук се складират всички докладни на катедра Компютърни системи и технологии

Приоритет: Нормален

Потребителска група: катедра Компютърни системи и технологии **(Този тип документ е достъпен само за членовете на потребителска група)**

Този тип документ също така може да бъде променян от членовете на потребителската група.

[Добави задача] [Промени] [Не го искам вече]

:: Файлове към този проект

[Добави файл]

:: Добави файл

Файл за качване: C:\dokladna_123.rtf Browse...

Описание: Докладна записка за електронни елементи

Файлът за качване трябва да е поне 2000 Кб.

Наистина ли да изпращам писмо при качване на файл до притежателя? ☑

Писмо до групата при качване на файл ☑

Качване

:: Детайли по тип документ

[Версия за печат]

Докладни кат. КСТ Напред!

Докладни кат. КСТ

Тук се складират всички докладни на катедра Компютърни системи и технологии

Приоритет: Нормален

Потребителска група: катедра Компютърни системи и технологии **(Този тип документ е достъпен само за членовете на потребителска група)**

Този тип документ също така може да бъде променян от членовете на потребителската група.

[Добави задача] [Промени] [Не го искам вече]

:: Файлове към този проект

dokladna_123.rtf (20480 байта) [Изтрий]

Качване: Administrator (2006-Август-16 23:22 +0300)

Докладна записка за електронни елементи

[Добави файл]

Fig. 3. The process for upload of filles

When is logged a user with administrative privileges it see the screen shown in fig. 4 and when is logged a user with users privileges it see the screen shown in fig. 5.

The user how is logged with users account was making:

- create the documents from templates;
- up-load of the instance documents;
- make archive from available documents;
- adding the new records in address book;

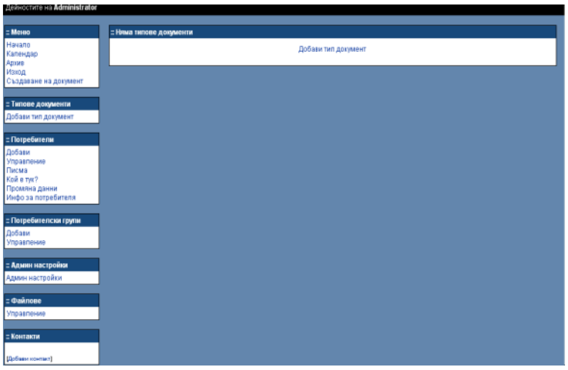
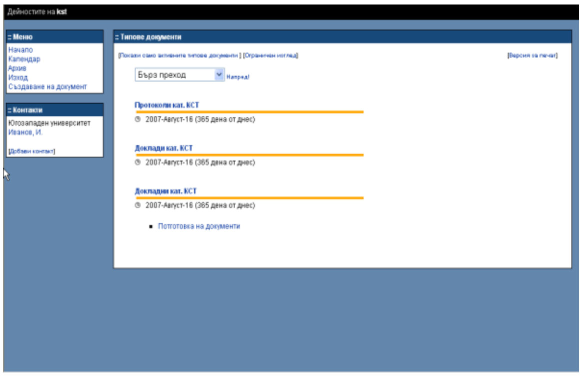


Fig. 4. Administrative start screen



Югозападен университет
Иванов, И.

:: Типове документи

[Покажи само активните типове документи] [Ограничен изглед] [Версия за печат]

Бърз преход Напред!

Протоколи кат. КСТ
2007-Август-16 (365 дена от днес)

Доклади кат. КСТ
2007-Август-16 (365 дена от днес)

Докладни кат. КСТ
2007-Август-16 (365 дена от днес)

- Потготовка на документи

Fig. 5. Users start screen

The user how is logged with administrative account was making the same thinks like users plus:

- changes in administrative accounts;
- create new accounts;
- assignment of usergroups; corresponding with departments;
- e-mails settings;

- tracking of the activity of system;
- founding of banks for documents;
- definition of task for usergroups;

Realization of this aims was making with creations of the PHP scripts. The uploading documents are saving in database. The systems save documents with unique numbers. This numbers are two – one how is output number and second is input number. FMNSwebDoc has possibility for tracing the road of document and in the every moment is available at with office is situated. This possibility is available only for administrative accounts.

4. Security

FMNSwebDoc has been coded with strong security measures. In practical use FMNSwebDoc security has proven to be very robust. Practical measures to enhance security include:

- Shift the uploaded file directory to outside the web server root. FMNSwebDoc can write to anywhere in the file system. Check that the web server has write permissions for the chosen directory. This prevents uses uploading files, then navigating to the upload directory with a browser and executing the file.
- Shift the config file out of the webserver directories. You will need to alter the file [FMNSwebDoc]/path_ config.php and add the new path to the include_path directive in php.ini.
- Shift the program files out of the webserver directories. You will need to alter the file [FMNSwebDoc] /path.php and add the new path to the include_path directive in php.ini.
- Use a non-root user for the database. The database users have minimal privileges, and not are able to create and/or delete tables.
- Make sure that Apache configuration allows .htaccess files to set <Limit> in FMNSwebDoc directory. FMN SwebDoc uses .htaccess files extensively to protect subdirectories.
- Make sure the file *[*FMNSwebDoc*] /config/config.php* is not world writable and not owned by the web server user.
- Use an SSL layer for access with FMNSwebDoc. Delete unnecessary files. For instance the *[*FMNSwebDoc*]/setup* directory and *[*FMNSwebDoc*]/update.php*. FMNSwebDoc is not tested nor supported on Microsoft IIS.

5. Conclusion

The present paper describes the developed web based system for documentation exchange in faculty. Based on analysis a decision was made for the operating system, web server, and dynamic pages platform is done. The functional diagram of the software and for the security design are proposed.

The real application of this product is present and running in web at address:

www.sec.swu.bg/pmf_doc/index.html.

References

[1] D. Sklar, Learning PHP 5, O'Reilly, 2004

[2] D. Sklar, A Trachtenberg, PHP Cookbook, O'Reilly, 2002

[3] R. Lerdorf, K. Tatroe, Programming PHP, O'Reilly, 2002

[4] P. DuBois, MySQL Cookbook, O'Reilly, 2002

[5] D. Dimitrov, Комуникационни системи в медицината, Изд. Технически Университет, София, 2005