

Features of developing a crm system for medical centers using mysql, php and javascript technologies

Gulara Aimbetova Nurymbetovna¹

Akhunov Farrukh Rustemovich²

Li Andrey Evgenievich³

Nukus branch of TUIT named after Muhammad al-Khwarizmi,
Nukus, Uzbekistan.

ABSTRACT

In the activities of any medical center, there is a need to automate the processes of interaction between patients and this center. In this regard, CRM (Customer Relationship Management) system MedId was developed, which is used to achieve the above goal. CRM system MedId was developed using MySQL, PHP and JavaScript technologies and has already been implemented in several medical centers.

ARTICLE INFO

Article history:

Received 24 December 2020

Revised form 16 January 2020

Accepted 06 Feb 2020

Keywords:

CRM, CRM system, MedID, MySQL, PHP, JavaScript, automation, medical center, database.

© 2019 Hosting by Central Asian Studies. All rights reserved.

1.INTRODUCTION

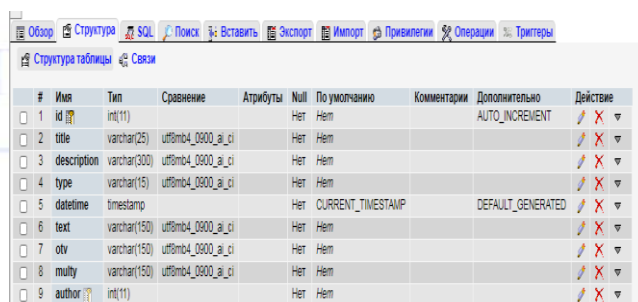
The aim of the work is software implementation of the CRM system. The bottom line is that after registration, the user identifier will only be his serial number. The elements of the chat also appeared in this system, since the correspondence between the support service and users is carried out instantly without reloading the page.

To solve the problem, the Apache software package was chosen, consisting of a bundle of PHP + MySQL. To achieve the effect of a dynamic site, scripts were written in JavaScript and its JQuery library. Also, AJAX technology was used to build an interactive interface consisting in the "background" exchange of data between the browser and the server.

Database design

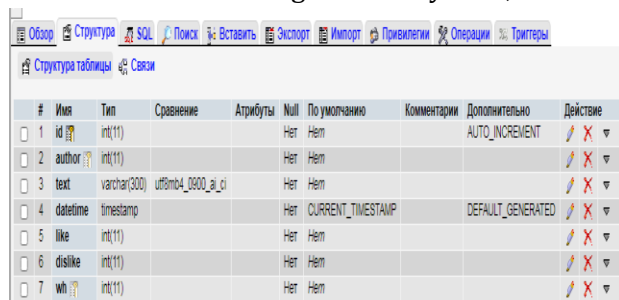
The phpMyAdmin system was used to develop the database (hereinafter DB). During the development process, the following was created:

theams - a table that stores information about all tasks in the system;



#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
1	id	int(11)			Нет	Нет		AUTO_INCREMENT	
2	title	varchar(25)	utf8mb4_0900_ai_ci		Нет	Нет			
3	description	varchar(300)	utf8mb4_0900_ai_ci		Нет	Нет			
4	type	varchar(15)	utf8mb4_0900_ai_ci		Нет	Нет			
5	datetime	timestamp			Нет	CURRENT_TIMESTAMP		DEFAULT_GENERATED	
6	text	varchar(150)	utf8mb4_0900_ai_ci		Нет	Нет			
7	otv	varchar(150)	utf8mb4_0900_ai_ci		Нет	Нет			
8	multy	varchar(150)	utf8mb4_0900_ai_ci		Нет	Нет			
9	author	int(11)			Нет	Нет			

Figure 1. Structure of the **theams** table for messages in the system;



#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
1	id	int(11)			Нет	Нет		AUTO_INCREMENT	
2	author	int(11)			Нет	Нет			
3	text	varchar(300)	utf8mb4_0900_ai_ci		Нет	Нет			
4	datetime	timestamp			Нет	CURRENT_TIMESTAMP		DEFAULT_GENERATED	
5	like	int(11)			Нет	Нет			
6	dislike	int(11)			Нет	Нет			
7	wh	int(11)			Нет	Нет			

Figure 2. Structure of the **mess** table favorite - a table for storing the id of those forums that the user has marked as favorites;

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
1	id	int(11)			Нет	Нет			
2	author	int(11)			Нет	Нет			

Figure 3. The structure of the favorite table **user** - a table that stores information about users registered in the system;

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
1	id	int(11)			Нет	Нет		AUTO_INCREMENT	
2	email	varchar(25)	utf8mb4_0900_ai_ci		Нет	Нет			
3	password	varchar(20)	utf8mb4_0900_ai_ci		Нет	Нет			
4	img	varchar(30)	utf8mb4_0900_ai_ci		Нет	Нет			
5	img_mini	varchar(30)	utf8mb4_0900_ai_ci		Нет	Нет			
6	userang	int(1)			Нет	Нет			

Figure 3. User table structure

connect is a table that stores the wishes of users.

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
1	id	int(11)			Нет	Нет		AUTO_INCREMENT	
2	text	varchar(200)	utf8mb4_0900_ai_ci		Нет	Нет			

Figure 4. Structure of the connect table

Site structure

The whole site can be divided into 8 pages:

- 1) The main page of the site.
- 2) New user registration page
- 3) User authorization.
- 4) Page of the list of tasks in the system.
- 5) The page for viewing a specific task.
- 6) Page for users.
- 7) Search page.
- 8) The page for creating a new task.

When developing the site, a library of functions was written:

- 1) Connection to the base.
- 2) Output of the site header.
- 3) Output of the site footer.
- 4) Getting data about the current user.
- 5) Declination of Russian words with numerals.
- 6) Correct date display.
- 7) Getting data about a user by his ID

Authentication system

The authentication system consists of user registration and authorization.

Figure 5. Authorization in the system

User registration lies in the reg.php module. When you open this module, the browser displays 4 fields: 2 for filling in the user's mailbox and 2 fields for entering the user's password. Duplication of input fields is necessary to prevent accidental errors when filling out data.

After the user enters and clicks on the register button, an AJAX request is sent to the reg.php script, which enters the new user into the table, in response, the script returns true if everything went well and the user was added to the database or html the error text, which is displayed in message.

User authorization is carried out using the login.php module, or through the panel displayed in the site header. The module, like the panel, consists of 2 fields: a field for entering a mailbox and a field for entering a password. When a user is authorized, data with a POST request is sent to the enter.php script, which checks the correctness of the login - password data entry.

If these data do not match, then a redirect (redirection) to the login.php page occurs, in which the type of error is transmitted using GET. If everything went well and the user logged in, then a session is created for him, in which there is 1 variable id, which stores the unique identifier of the user, cookies are set or overwritten, and a redirect (redirection) to the theams.php page occurs.

Home page of the site

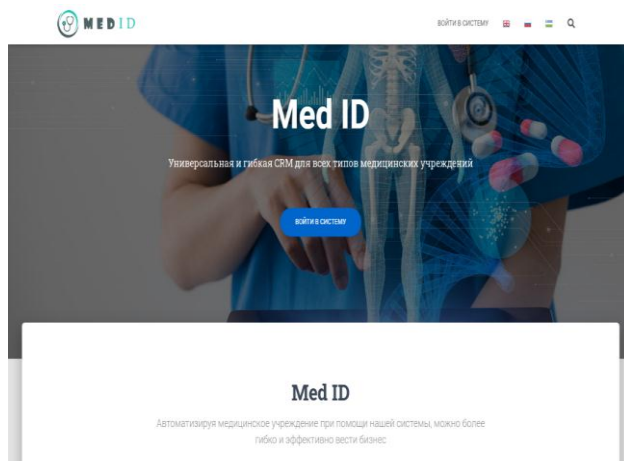


Figure 6. Home page of the site index.php

The main page (index.php) is only visible to unregistered users, as the logged in user is always redirected to the theams.php page.

When opening the index.php page, it first searches for the user's cookies. If cookies are found, then we start checking them. If the check is successful, then we start a session on the user.

Otherwise, we display the page for an unregistered user.

System task list page

After registration, the script sends the user to this particular page (theams.php). On this page, if the user is not registered, only a list of topics that have been created by users of the site is displayed. But if the user is registered, then some of the user's functions are displayed in the upper part, such as: adding a new topic, viewing new posts on the forum, and switching between favorites and the usual list of topics.

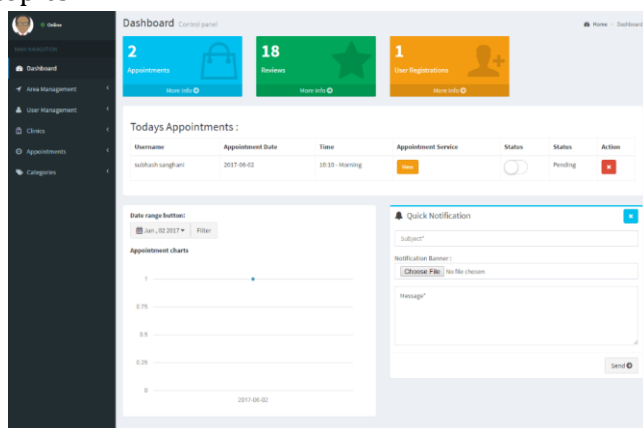


Figure 7. theams.php task list page

The page for adding a new task to the system

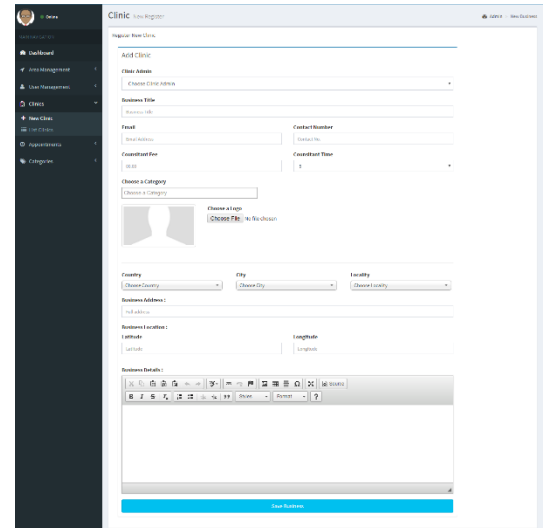


Figure 8. The page for adding a new theme add.php

Adding a new topic to the forum (add.php) is done in 2 steps.

Step 1. It displays what type of topic the user wants to create, either a regular topic or a poll.

Step 2. After choosing the type of topic, the user begins to enter the necessary data, depending on which type of topic he has chosen. For example, for a survey, new required fields are added: fields for entering answer options.

The page for viewing tasks in the system

To open this page (theam.php), you need to pass the required post parameter by the GET 1 method, this parameter tells the module which page to display information about.

Then, when opening this page, it pulls the topic type from the database, and if the type is interview, then after the title and description of the topic, the topic survey is displayed. If the multy field is true, then a poll with a multiple choice option will be displayed, otherwise a poll with 1 answer option will be displayed.

Further, all messages are taken from the table `mess` and displayed as a list.

After displaying messages, if the user is registered, the panel for sending messages on the forum is displayed.

Sending messages, as well as registration, is carried out using AJAX technology to the php / address.php script. This script, if a new message has been successfully added, returns the page on which this message will be displayed. If this page does not coincide with the current one, then a redirect

(redirection) to the received page occurs, and messages are output to the current page using AJAX.

If the script was unable to add a message to the database, then the error HTML is returned, which is displayed in the message.

Also, at the moment of opening the page, the recursive process of sending AJAX requests to the php / newmess.php script begins. This script returns 0 - if there are no new messages on this page. Or it returns N - the number of new messages and the message "Added N new messages" appears in the lower right corner.

If the current page is the same as the page for the new message, then when you click on this message by means of AJAX all new messages will be displayed. Otherwise, a redirect (redirect) to the page with a new message will occur and the browser will automatically, if necessary, scroll to this particular message.

Search page

The search page (search.php) can be accessed from any page of the site, to go to this page, you just need to enter a search query in the site header.

Site search is carried out using the keyword LIKE in the SQL query. The search goes through all forums on the site.

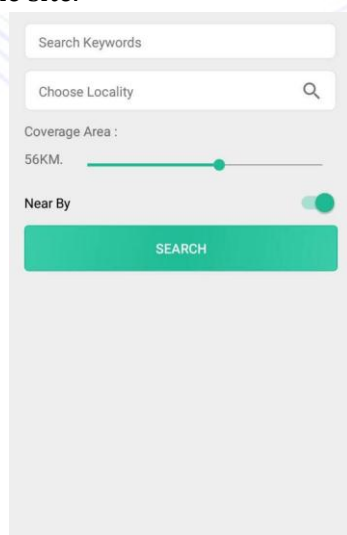


Figure: 9. Search page search.php

User page

To get to the user's page, you need to pass the id parameter using the GET method. This parameter tells the module (user.php) information about which user to display.

This page, if the user is not authorized, displays the number of messages, the number of created topics, a list of these topics, the last message sent by the user, etc.

But, if the user is authorized and the id matches the unique identifier of the user, then this module also displays the settings for the profile. These settings include: changing the password, changing the mailbox, the ability to delete or upload an image, and you can also change the number of displayed topics per page.

When a mailbox is changed, 2 letters are sent to both mailboxes to 2 different mailboxes with unique links, following which the mailbox will change. When changing the password, you will need to enter the old password and the new one. After entering, it will be checked for correctness and if the result is successful, the password will change.

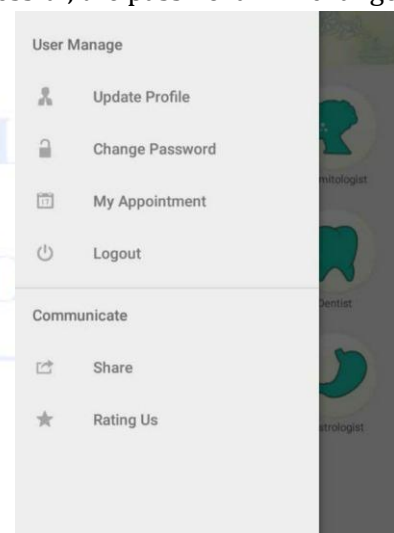


Figure: 10. User page user.php

Conclusion. The result of the work done on the project was a software product, which is a database in the form of a set of files, containing tables related by key fields. The data in the database has been normalized to thirds normal form. For the created database, a universal request form has been developed that allows you to enter data into the database, sort it and receive a report on the movement of inventory items. The report format is a Microsoft Office Word 2003 document or higher.

In the analytical part, the information and technical infrastructure of the enterprise was analyzed, an action plan was drawn up for its subsequent modernization.

In the practical part, a finished software product was received.

As a result, the finished system can be used at the following web addresses: www.medid.uz and www.crm.medid.uz

REFERENCES

1. А.Я. Архангельский "Программирование PHP" Москва 2009г.
2. А.Я. Архангельский "Приемы программирования на PHP" Москва 2012г.
3. Стив Тейксера, Ксавье Пачеко "PHP+MySQL Руководство разработчика" Москва 2008г.
4. Фень Юань "web программирование" Москва 2012г.
5. Ручкин, В.Н. Архитектура компьютерных сетей. / В.Н. Ручкин, В.А. Фулин - Диалог-МИФИ, 2008. - 240 с.
6. Описание стандартов семейства IDEF www.odef.org / [Электронный ресурс]. – Internet recourse режим доступа www.odef.org
7. Столингс, В. Основы защиты сетей. Приложения и стандарты. / В. Столингс - Вильямс, 2012. - 432 с.