

**ГОСУДАРСТВЕННЫЙ КОМИТЕТ СВЯЗИ,  
ИНФОРМАТИЗАЦИИ И ТЕЛЕКОММУНИКАЦИОННЫХ  
ТЕХНОЛОГИЙ РЕСПУБЛИКИ УЗБЕКИСТАН**

**ФЕРГАНСКИЙ ФИЛИАЛ ТАШКЕНТСКОГО  
УНИВЕРСИТЕТА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ**

**На правах рукописи**

**УДК 66.012-52**

**Б.ХОМИДОВ**

**“Сравнительный анализ технических и методических аспектов  
информационно-допризывной системы в Ферганской области и  
создания его программного обеспечения”**

**5А330201 – Компьютерные системы и их программное  
обеспечение.**

**Диссертационный проект на соискание степени магистра**

**Научный руководитель:**

**зам. нач. производства АС-АК72М:**

**Измирян А.В.**

## **Аннотация**

**Объектом исследования является возможность разработки и создание базы данных для разработки алгоритма и описание работы программы обработки данных призывника для военкомата.**

**При создании базы данных были изучены, рассмотрены программные пакеты по созданию базы данных (PHP, My SQL, NetObjects Fussion, Home Site, Adobe Dreamweaver CS4).**

**Были рассмотрены различные системы управления содержимым, обеспечивающие доступ к информации в сети Internet и удобное редактирование базы данных. Из многообразия существующих в настоящее время была выбрана наиболее оптимальная для работы с Web-приложениями, отвечающая всем требованиям разработчика.**

**Результатом данной работы является разработка информационной структуры, разработка алгоритма и описание работы программы обработки данных призывника для военкомата дизайна и информационное наполнение web-базы данных, который предоставляет всю необходимую информацию.**

**Созданный программы базы данных позволит получить исчерпывающую информацию о представленных материалах, разрешит просмотр новостей, будет поддерживать аутентификацию пользователей по логину и паролю, а также позволит получать необходимую информацию о текущих и запланированных призывников в режиме реального времени.**

**В качестве дальнейшего совершенствования базы данных представляется возможным разработка модулей доступа к системам дистанционного образования. Так же возможна доработка интерфейса базы данных с целью дальнейшего повышения его информативности, привлекательности и удобства.**

**Поставленные цели и задачи выполнены и разработанный модуль вполне можно применять. Программа установлена Ферганком городском в настоящее время проходит обработку в Ферганком городском военкомате имеются акт о внедрении программного продукция.**

## **СОДЕРЖАНИЕ**

|  |           |
|--|-----------|
| <b>ВВЕДЕНИЕ .....</b>  | <b>6</b>  |
| <b>I ГЛАВА АНАЛИЗ ТЕХНИЧЕСКИХ И МЕТОДИЧЕСКИХ АСПЕКТОВ ПРОГРАММНЫХ СРЕДСТВ.....</b> | <b>10</b> |
| <b>1.1 ВЫБОР ПРОГРАММНЫХ СРЕДСТВ .....</b>   | <b>10</b> |
| <b>1.2 ПРЕИМУЩЕСТВА ИСПОЛЬЗОВАНИЯ PHP В СРАВНЕНИИ С ДРУГИМИ ЯЗЫКАМИ .....</b>      | <b>11</b> |
| <b>1.3 СЕРВЕР БАЗ ДАННЫХ MYSQL И ЕГО ОСОБЕННОСТИ.....</b>                          | <b>15</b> |
| <b>1.4 ВЫБОР СРЕДСТВА ПРОЕКТИРОВАНИЯ WEB-САЙТА .....</b>                           | <b>16</b> |
| <b>1.5 УСТАНОВКА И НАСТРОЙКА АППАРАТНО-ПРОГРАММНОГО КОМПЛЕКСА.....</b>             | <b>18</b> |
| <b>1.6 WEB - СЕРВЕР APACHE.....</b>  | <b>18</b> |
| <b>1.7 УСТАНОВКА PHP И НАСТРОЙКА APACHE ДЛЯ РАБОТЫ С PHP</b>                       | <b>22</b> |
| <b>1.8 УСТАНОВКА MYSQL .....</b>   | <b>24</b> |
| <b>1.9 РАБОТА В MACROMEDIA DREAMWEAVER MX 2013 .....</b>                           | <b>25</b> |
| <b>II ГЛАВА ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ.....</b>                                    | <b>32</b> |
| <b>ИНФОРМАЦИОННО-ДОПРИЗЫВНОЙ СИСТЕМЫ.....</b>                                      | <b>32</b> |
| <b>2.1 РАЗРАБОТКА БАЗЫ ДАННЫХ .....</b>  | <b>32</b> |
| <b>2.2 Разработка пользовательского интерфейса</b>                                 | <b>32</b> |
| <b>2.3 ФАЙЛОВАЯ СТРУКТУРА БАЗЫ ДАННЫХ .....</b>                                    | <b>33</b> |
| <b>2.4 ВХОД В СИСТЕМУ MYSQL .....</b>  | <b>34</b> |
| <b>2.5 СОЗДАНИЕ ТАБЛИЦ БАЗЫ ДАННЫХ .....</b>                                       | <b>35</b> |
| <b>2.6 ФОРМИРОВАНИЕ СТРАНИЦ БАЗЫ ДАННЫХ .....</b>                                  | <b>43</b> |
| <b>2.7 ИСПОЛЬЗОВАНИЕ ПЕРЕМЕННЫХ СЕАНСА .....</b>                                   | <b>50</b> |
| <b>2.8 ИСПОЛЬЗОВАНИЕ СОOKIE - НАБОРОВ В СЕАНСЕ .....</b>                           | <b>51</b> |
| <b>2.8.1 РЕАЛИЗАЦИЯ УПРАВЛЕНИЯ ПРОСТЫМ СЕАНСОМ.....</b>                            | <b>52</b> |

|  |           |
|--|-----------|
| <b>2.8.2 ЗАПУСК СЕАНСА.....</b>  | <b>53</b> |
| <b>2.8.3 РЕГИСТРАЦИЯ ПЕРЕМЕННЫХ СЕАНСА.....</b>                                    | <b>53</b> |
| <b>2.8.4 ИСПОЛЬЗОВАНИЕ ПЕРЕМЕННЫХ СЕАНСА .....</b>                                 | <b>54</b> |
| <b>2.8.5 ОТМЕНА РЕГИСТРАЦИИ ПЕРЕМЕННЫХ И ЗАВЕРШЕНИЕ СЕАНСА.....</b>                | <b>56</b> |
| <b>2.8.6 ИСПОЛЬЗОВАНИЕ ПЕРЕМЕННЫХ СЕАНСА В ПРИЛОЖЕНИИ ...</b>                      | <b>56</b> |
| <b>2.9 ЗАПУСК БАЗЫ ДАННЫХ И ОПИСАНИЕ РАБОТЫ С НИМ .....</b>                        | <b>58</b> |
| <b>III РАЗРАБОТКА АЛГОРИТМА И ОПИСАНИЕ РАБОТЫ ПРОГРАММЫ ОБРАБОТКИ ДАННЫХ .....</b> | <b>59</b> |
| <b>3.1 ВЫБОР МОДЕЛИ ПРОЕКТИРОВАНИЯ.....</b>  | <b>59</b> |
| <b>3.2 ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ К ПРИЛОЖЕНИЮ .....</b>                               | <b>61</b> |
| <b>3.3 ЭТАПЫ РАЗРАБОТКИ ПРИЛОЖЕНИЯ .....</b>                                       | <b>61</b> |
| <b>3.4 ВЫБОР МОДЕЛИ СОЗДАНИЯ САЙТА .....</b>                                       | <b>63</b> |
| <b>3.5 СОЗДАНИЕ МЕНЮ ПРОГРАММЫ.....</b>  | <b>66</b> |
| <b>3.6 СОЗДАНИЕ БЛОКА ВЫВОДА ЗАПИСАННЫХ ПРИЗЫВНИКОВ.....</b>                       | <b>78</b> |
| <b>3.7 ОПИСАНИЕ АЛГОРИТМА ПРОГРАММЫ.....</b>                                       | <b>81</b> |
| <b>3.8 ОФОРМЛЕНИЕ СТРАНИЦЫ .....</b>   | <b>85</b> |
| <b>3.9 ОПИСАНИЕ РАБОТЫ ПРОГРАММЫ .....</b>   | <b>89</b> |
| <b>ЗАКЛЮЧЕНИЕ .....</b>  | <b>93</b> |
| <b>СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ.....</b>                                       | <b>95</b> |
| <b>ПРИЛОЖЕНИЯ .....</b>  | <b>98</b> |

## ВВЕДЕНИЕ

### **Объем и структура диссертации.**

**Диссертационная работа написана на русском языке. Она состоит из введения, 3х глав, заключения испискаиспользованной литературы составляющий 28 наименований и 5-ти дополнительной литературы. В первых 2 –х приложениях приводятся, тело написанной программы по различным разделам программного обеспечения и в приложении приводится список приведённых сокращений.**

**С появлением Web-технологии компьютер начинают использовать совершенно новые слои населения Земли. Можно выделить две наиболее характерные группы, находящиеся на разных социальных полюсах, которые были стремительно вовлечены в новую технологию, возможно, даже помимо их собственного желания. С одной стороны, это были представители элитарных групп общества – руководители крупных организаций, президенты банков, топ- менеджеры, влиятельные государственные чиновники и т.д. С другой стороны, это были представители широчайших слоев населения – домохозяйки, пенсионеры, дети.**

**При появлении технологии Web компьютеры повернулись лицом к этим двум совершенно противоположным категориям потенциальных пользователей. Элиту объединяла одна черта – в силу высочайшей ответственности и практически стопроцентной занятости “большие люди” никогда не пользовались компьютером; типичной была ситуация, когда с**

компьютером работал секретарь. В какой-то момент времени они поняли, что компьютер им может быть полезен, что они могут результативно использовать то небольшое время, которое можно выделить на работу за компьютером. Они вдруг поняли, что компьютер – это не просто модная и дорогая игрушка, но инструмент получения актуальной информации для бизнеса. При этом им не нужно было тратить сколько-нибудь заметного времени, чтобы освоить технологию работы с компьютером (по сравнению с тем, как это было раньше).

Спектр социальных групп, подключающихся к сети Интернет и ищущих информацию в WWW, все время расширяется за счет пользователей, не относящихся к категории специалистов в области информационных технологий. Это врачи, строители, историки, юристы, финансисты, спортсмены, путешественники, священнослужители, артисты, писатели, художники. Список можно продолжать бесконечно. Любой, кто ощутил полезность и незаменимость Сети для своей профессиональной деятельности или увлечений, присоединяется к огромной армии потребителей информации во «Всемирной Паутине».

Web-технология полностью перевернула наши представления о работе с информацией, да и с компьютером вообще. Оказалось, что традиционные параметры развития вычислительной техники – производительность, пропускная способность, емкость запоминающих устройств – не учитывали главного «узкого места» системы – интерфейса с человеком. Устаревший

**механизм взаимодействия человека с информационной системой сдерживал внедрение новых технологий и уменьшал выгоду от их применения. И только когда интерфейс между человеком и компьютером был упрощен до естественности восприятия обычным человеком, последовал беспрецедентный взрыв интереса к возможностям вычислительной техники.**

**С развитием технологий гипертекстовой разметки в Интернете стало появляться всё больше сайтов, тематика которых была совершенно различной – от сайтов крупных компаний, повествующих об успехах компании и её провалах, до сайтов маленьких фирм, предлагающих посетить их офисы в пределах одного города.**

**Развитие Интернет-технологий послужило толчком к появлению новой ветки в Интернете – Интернет- форумов. Стали появляться сайты, и даже целые порталы, на которых люди со всех уголков планеты могут общаться, получать ответы на любые вопросы и, даже, заключать деловые сделки.**

**Объект исследования. Разработка программного обеспечения регистрации информационно-допризывной системы.**

**Цель и задачи исследования Сравнительный анализ технических и методических аспектов информационно-допризывной системы в Ферганской области и создания его программного обеспечения. Основной задачей исследования этой магистерской диссертации является разработка программного средства для военкомата.**

**Метод исследования.**

**Для метода исследование были выбраны WEB технологии. В том числе для разработки программного обеспечения был использован следующий язык программирования: PHP, а также JAVA (сценария). При разработки базы данных, использовано технология MySQL.**

**Проблема исследования.**

**Проблемой исследование, разработка системы с помощью которой можно было бы уменьшить использование бумажных средств и повышения эффективности рабочего процесса.**

**Научная новизна исследование.**

**Развитие информационно-коммуникационных технологий (ИКТ) в Узбекистане, являющееся важнейшим фактором поднятия благосостояния и экономического роста, становится одним из основных приоритетов государственной политики. Данная магистерская диссертация направлена на сравнительный анализ технических и методических аспектов, информационно-допризывной системы в ферганской области и создание программного обеспечения.**

# I ГЛАВА. АНАЛИЗ ТЕХНИЧЕСКИХ И МЕТОДИЧЕСКИХ АСПЕКТОВ ПРОГРАММНЫХ СРЕДСТВ

## 1.1 ВЫБОР ПРОГРАММНЫХ СРЕДСТВ

Современные программы базы данных становятся все более сложными и все более перегружаются логикой. Раньше производительность таких приложений определялась, в основном, скоростью работы того или иного SQL-сервера и тем, существует ли для него достаточно эффективная реализация драйвера доступа к SQL-серверу для выбранного языка программирования. Это объясняется тем, что первое поколение веб-сайтов просто читало и писало информацию в базы данных. Пользователей при этом было относительно немного. Таким образом, время на отработку SQL-запроса составляло 70-90% от общего времени обработки HTTP-запроса.

С повышением требований к масштабируемости (увеличение количества пользователей) и наращиванием логики приложения требования к языку программирования и среде выполнения существенно возрастают. К этому следует также прибавить, что относительно недавно веб-сайты перешли из мира Интернет в мир корпоративных приложений. Это снова повысило требования к эффективности среды выполнения.

В настоящее время для создания веб-сайтов существует множество различных языков программирования. Самые популярные из них - PHP, Perl, C# (DOT. NET), Java2. Классическая технология ASP становится историей, поскольку

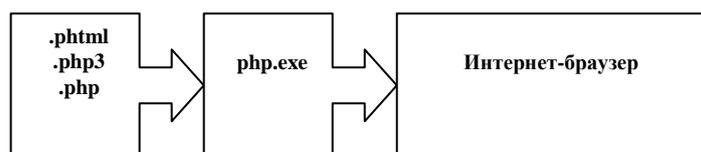
**ASP DOT. NET практически вытеснило эту технологию на Windows-серверах в новых приложениях.**

**Для разработки в работе выбран язык PHP в связке с сервером баз данных MySQL. Для их работы требуется установка веб-сервера Apache. В качестве среды разработки выбрано средство создания web-узлов Macromedia Dreamweaver MX 2004.**

## **1.2 ПРЕИМУЩЕСТВА ИСПОЛЬЗОВАНИЯ PHP В СРАВНЕНИИ С ДРУГИМИ ЯЗЫКАМИ**

**PHP - это инструмент объектно-ориентированного программирования с открытым кодом (open source), язык серверных скриптов (server scripting language), встраиваемый в HTML, который интерпретируется и выполняется на сервере [13].**

**PHP является препроцессором HTML. Его работа построена по схеме, показанной на рисунке 2.4**



**Рисунок 2.4 - Схема работы препроцессора PHP**

**До того, как сервер "отдаст" файл браузеру, его просматривает препроцессор-интерпретатор. Для того, чтобы это происходило, файлы, которые подвергаются обработке препроцессором, должны иметь определенное расширение (обычно это. phtml или. php, но эти значения можно поменять) и содержать (хотя**

это не обязательное требование) код для препроцессора. Перед отправкой страницы PHP-код проигрывается на сервере и браузеру выдается результат в виде опять таки HTML-страницы, которая может сильно отличаться от той, что хранится на сервере. Обычные же страницы, имеющие расширение .html/.htm web-сервер будет отправлять браузеру без какой-либо обработки.

Основное отличие от CGI-скриптов, написанных на других языках, типа Perl или C - это то, что в CGI-программах вы сами пишете выводимый HTML-код, а, используя PHP - вы встраиваете свою программу-скрипт в готовую HTML-страницу, используя открывающий и закрывающий теги (в примере `<? php и? >`).

PHP называется языком серверных скриптов в отличие от JavaScript/Jscript/VBScript, которые являются языками клиентских скриптов. Это значит, что PHP-скрипт выполняется на сервере, а клиенту передается результат его работы, тогда как в JavaScript-код полностью передается на клиентскую машину и только там выполняется браузером.

Пользователи MS Internet Information Server найдут, что PHP очень похож на Active Server Pages (ASP), а пользователи Java скажут, что PHP похож на Java Server Pages (JSP). Еще некоторыми аналогами PHP являются языки ColdFusion и embPerl. Все эти языки позволяют размещать код, выполняемый на Web-сервере, внутри HTML-страниц. В результирующей web-странице нет и следа PHP-кода.

**На PHP можно сделать все, что можно сделать с помощью CGI-программ. Например: обрабатывать данные из форм, генерировать динамические страницы, получать и посылать куки (cookies).**

**Кроме этого в PHP включена поддержка многих баз данных (databases), что делает написание Web-приложений с использованием БД до невозможности простым.**

**Вот неполный перечень поддерживаемых БД: Adabas D, InterBase, Solid, dBase, mSQL, Sybase, Empress, MySQL, Velocis, FilePro, Oracle, Unix dbm, Informix, PostgreSQL, ODBC.**

**Вдобавок ко всему PHP понимает протоколы IMAP, SNMP, NNTP, POP3 и даже HTTP, а также имеет возможность работать с сокетами (sockets) и общаться по другим протоколам.**

**Разработчикам web-приложений нет необходимости говорить, что web-страницы - это не только текст и картинки. Достойный внимания сайт должен поддерживать некоторый уровень интерактивности с пользователем: поиск информации, продажа продуктов, конференции и т.п. До недавних пор все это традиционно реализовалось CGI-скриптами, написанными на Perl. Но оказалось, что CGI-скрипты очень плохо масштабируемы. Каждый новый вызов CGI-скрипта, требует от ядра порождения нового процесса, а это занимает процессорное время и тратит оперативную память. PHP предлагает другой вариант - он работает как часть web-сервера,**

и этим самым похож на ASP от Microsoft или ColdFusion от Allaire.

Синтаксис PHP очень похож на синтаксис C или Perl. Люди, знакомые с программированием, очень быстро смогут начать писать программы на PHP. В этом языке нет строгой типизации данных и нет необходимости в действиях по выделению/освобождению памяти.

Программы, написанные на PHP, читаются достаточно легко. В отличие от Perl-программ PHP-код легко зрительно прочитать и понять.

В дополнение к своей бесплатности (хотя MySQL требует приобретения лицензии при использовании ее в коммерческих целях) связка PHP-MySQL является кросс-платформенной. Это значит, что вы можете, работая в Windows, разрабатывать приложения, предназначенные для работы под Unix. Кроме того, PHP может работать как внешний CGI-процесс, либо как обычный интерпретатор скриптов, либо как модуль, подключаемый к web-серверу Apache или IIS.

И наконец, так как данный продукт разрабатывается совместными усилиями, существует огромное количество документации и списков рассылки, к которым можно обратиться в случае возникновения каких-либо вопросов.

Найденные ошибки исправляются достаточно быстро, ваши предложения и замечания всегда выслушают, рассмотрят, и если они окажутся ценными - реализуют в новой версии.

### 1.3 СЕРВЕР БАЗ ДАННЫХ MySQL И ЕГО ОСОБЕННОСТИ

**MySQL - небольшой, компактный многопоточный сервер баз данных [13]. MySQL характеризуется большой скоростью, устойчивостью и легкостью в использовании.**

**MySQL был разработан компанией ТсХ для внутренних нужд, которые заключались в быстрой обработке очень больших баз данных. Компания утверждает, что использует MySQL с 1996 года на сервере с более чем 40 БД, которые содержат 10,000 таблиц, из которых более чем 500 имеют более 7 миллионов строк. MySQL является идеальным решением для малых и средних приложений. Исходные тексты сервера компилируются на множестве платформ. Наиболее полно возможности сервера проявляются на Unix-серверах, где есть поддержка многопоточности, что дает значительный прирост производительности. В варианте под Windows, MySQL может запускаться как сервис Windows NT или как обычный процесс на Windows 95/98.**

**MySQL-сервер является бесплатным для некоммерческого использования.**

**MySQL поддерживает язык запросов SQL в стандарте ANSI 92, и кроме этого имеет множество расширений к этому стандарту, которых нет ни в одной другой СУБД.**

**Краткий перечень возможностей MySQL.**

**Поддерживается неограниченное количество пользователей, одновременно работающих с базой данных;**

**Количество строк в таблицах может достигать 50 млн;**

**Быстрое выполнение команд. Возможно MySQL самый быстрый сервер из существующих;**

**Простая и эффективная система безопасности.**

**MySQL действительно очень быстрый сервер, но для достижения этого разработчикам пришлось пожертвовать некоторыми требованиями к реляционным СУБД. В MySQL отсутствуют:**

**Не реализована поддержка транзакций. Взамен предлагается использовать LOCK/UNLOCK TABLE;**

**Нет поддержки триггеров и хранимых процедур;**

**Нет поддержки внешних ключей.**

**По словам создателей именно эти пункты дали возможность достичь высокого быстродействия. Их реализация существенно снижает скорость сервера. Эти возможности не являются критичными при создании web-приложений, что в сочетании с высоким быстродействием и малой ценой позволило серверу приобрести большую популярность.**

#### **1.4 ВЫБОР СРЕДСТВА ПРОЕКТИРОВАНИЯ WEB-САЙТА**

**На сегодняшний момент существует два мощных средства создания веб-сайтов - это последние версии FrontPage 2003 компании Microsoft и Dreamweaver MX 2004 компании Macromedia [14].**

**Программный пакет FrontPage долгое время развивался и совершенствовался фирмой Microsoft. Его версия FrontPage 2000 отличается от всех предыдущих тем, что она впервые интегрирована в структуру пакета программ Microsoft Office.**

Вероятно, это объясняется тем обстоятельством, что проектирование веб-сайтов и мультимедийных электронных изданий из задачи для профессионалов перешли в сферу офисной деятельности. FrontPage 2002 вновь поставляется отдельно от Microsoft Office XP.

FrontPage специализирован на подготовке электронных изданий, в частности - на разработке web-узлов целиком. Он поддерживает фреймовую структуру, ориентирован на простоту и удобство создания гиперссылок, обеспечивает единый стиль оформления различных web-страниц, оптимизирует скорость их загрузки. Таким образом, пакет отличается продуманностью реализации как эстетических, так и технических аспектов создания электронных изданий и позволяет, в какой-то степени, автоматизировать процесс проектирования. Неудобством в работе является автоматическая вставка не всегда нужных тэгов.

Программный пакет Dreamweaver, как и FrontPage, предназначен для проектирования различных сайтов, в том числе и сайтов издательств, одной из задач которых является распространение электронных изданий. Он пригоден как для создания небольших, так и очень больших сайтов. В первом случае разработкой занимается единственный специалист - web-мастер. В последнем случае над проектом работает целый коллектив, включающий администратора проекта, web-дизайнера, HTML-верстальщика, web-программиста, корректора, менеджера рассылок. Многие специалисты

утверждают, что профессиональные web-дизайнеры в своей работе чаще всего используют Adobe Photoshop для подготовки графики и Macromedia Dreamweaver - для дизайна отдельных web-страниц. Особенностью данного программного средства является хорошая поддержка языка PHP, который используется в проекте для программирования и автоматическая или ручная синхронизация измененных данных с удаленным сервером. Эти особенности, а также простота и быстрота работы программы во многом обуславливают выбор именно этого программного продукта в качестве средства проектирования разрабатываемого web-приложение.

#### 1.5 УСТАНОВКА И НАСТРОЙКА АППАРАТНО-ПРОГРАММНОГО КОМПЛЕКСА

Для разработки системы используется комплекс программ. В качестве веб-сервера выбран Apache версии 2.0.35, в качестве языка программирования PHP 4.1.2, в качестве сервера баз данных MySQL 4.0.21. Для разработки и тестирования приложения используется Macromedia Dreamweaver MX 2004.

#### 1.6 ВЕБ - СЕРВЕР APACHE

Функции веб-сервера выполняет программа Apache. Когда браузер пользователя подключается к веб-серверу и посылает заголовок GET (это просьба передать файл), его запрос обрабатывает Apache. Apache проверяет, есть ли указанный в заголовке GET файл, и, если есть, отправляет его вместе с заголовками браузеру.

**Apache** - это стандарт веб-сервера в Интернете. Его основным конкурентом является **IIS (Internet Information Server)** от **Microsoft**, который работает под **Windows**. **Apache**, хотя и существуют его версии под **Windows**, преимущественно устанавливается на **Unix-подобных операционных системах - Linux и FreeBSD**. Нужно также заметить, что большинство хостеров используют именно **Apache**, а не **IIS**. **IIS** устанавливается в основном на корпоративных серверах, работающих под **Windows**.

Сервер **Apache** просто необходим - без него мы не сможем протестировать веб сценарии **PHP**. Конечно, можно тестировать и без сервера, но тогда нужно будет использовать сервер хостера. В этом случае затраты на Интернет сильно возрастут. Для установки **Apache** для **Windows** нужно посетить сайт <http://www.apache.org> и загрузить версию **Apache** для **Windows**. Установка сервера проблем не вызывает: нужно просто запустить программу установки и выбрать тип установки **Typical**. Еще потребуется указать каталог, в который будет производиться установка. По умолчанию будет установлен в каталог "**C:\Program Files\Apache Group\Apache**". По окончании установки **Apache** необходимо настроить. По умолчанию корневым каталогом для **HTML-документов** будет **C:\Program Files\Apache Group\Apache\htdocs**. Такое расположение не очень удобно, поэтому создадим каталог **f:\home\localhost**, в котором будут находиться файлы. В этом же

каталоге создадим два подкаталога - `cgi` и `html`. В первом будут находиться CGI-приложения, а во втором - HTML-файлы.

Затем нужно открыть в любом текстовом редакторе файл `C:\Program Files\Apache Group\Apache\conf\httpd.conf` и отредактировать следующие директивы:

`ServerName` - установить имя сервера. При этом, директива `ServerName` по умолчанию закомментирована, то есть игнорируется сервером. Поэтому необходимо раскомментировать эту директиву, убрав перед ней значок `#`.

`DocumentRoot` - установим значение `C:/www/html`:

```
DocumentRoot c:/www/html
```

Таким образом, укажем каталог, в котором будут размещаться HTML-файлы.

Установим директивы `ScriptAlias` так:

```
ScriptAlias/cgi/"c:/www/cgi/"
```

```
ScriptAlias/cgi-bin/"c:/www/cgi/"
```

Благодаря этому создадим два псевдонима для каталога `c:/www/cgi`. Таким образом, если будет указан путь `http://localhost/cgi` или `http://localhost/cgi-bin`, то на самом деле обращение будет к каталогу `c:/www/cgi`.

Секцию `<Directory "C:/Program Files/Apache Group/Apache/htdocs">` заменим на следующую:

```
<Directory "c:/www">
```

```
Options Indexes Includes AllowOverride All Allow from all  
</Directory>
```

Таким образом, зададим ряд настроек по умолчанию для каталога `c:\www` и всех его подкаталогов. А

**именно: автоматическую генерацию списка содержимого каталога при просмотре его в браузере, поддержку SSI, а также возможность использовать файлы. access для индивидуальной настройки каталогов.**

**Убедимся, что есть директива AddHandler:**

**AddHandler cgi-script. bat. exe. cgi**

**Эта директива указывает Apache на то, что файл с расширениями. bat. exe и. cgi следует воспринимать как CGI-модули.**

**Установим параметры SSI:**

**Addtype text/html. shtml**

**AddHandler server-parsed. shtml. html. htm** Благодаря этому файлы с указанными расширениями будут обрабатываться SSI. Необходимо сохранить отредактированный файл.

**Теперь протестируем работу Apache. Запуск:**

**C: \Program Files\Apache Group\Apache\apache. exe**

**При успешном запуске появится надпись**

**Apache/2.0.35 (Win32) running...**

**Теперь можно проверить корректность работы сервера. Запустим Internet Explorer или любой другой браузер и введем: http://localhost. Вы должны увидеть стартовую страницу Apache. Теперь проверим CGI и SSI. Для этого создадим файл test. bat и поместим его в каталог c: \www\cgi:**

**@echo off**

**echo Content-type: text/html**

**echo.**

**echo.**

**echo Hello!**

**В окне браузера введите <http://localhost/cgi/test.bat> -увидите слово Hello! Для проверки SSI нужно создать файл test.html и поместить его в каталог c:\www\html:**

**SSI Test<br>**

**<! -tinclude virtual="/index.html" ->**

**SSI (Server Side Include) - включение со стороны сервера, позволяет включать в Web-страницу код HTML, находящийся в других файлах.**

**В окне браузера при вызове файла test.html вы увидите SSI Test, а затем - стартовую страницу Apache.**

## **1.7 УСТАНОВКА PHP И НАСТРОЙКА APACHE ДЛЯ РАБОТЫ С PHP**

**После установки Apache установим PHP 4. Версию PHP 4 для Windows можно скачать по адресу <http://www.php.net> (последняя версия PHP доступна по адресу <http://www.php.net/downloads.php>).**

**Запустим программу установки, которая по умолчанию установит PHP 4 в каталог C:\Program Files\PHP 4\.** Однако в процессе установки можно выбрать другой каталог.

**Сразу после установки необходимо открыть файл php.ini (он будет в каталоге, куда вы установили PHP) и отредактировать следующие параметры:**

**[mail function] ; For Win32 only. SMTP = localhost**

**sendmail\_from = me@localhost.com**

**Параметр SMTP задает имя SMTP-сервера. По умолчанию в качестве SMTP-сервера используется локальный компьютер - это хорошо для Linux, но в Windows вам нужно указать SMTP-сервер провайдера, потому что вряд ли у вас будет установлен почтовик под Windows. Если вы не сделаете этого, функция mail(), отправляющая сообщения, работать не будет.**

**После этого откроем файл httpd.conf (файл конфигурации Apache) и найдем в нем строку:**

**AddType application/x-httpd-php php**

**Убедимся, что данная строка не закомментирована, то есть перед ней не стоит значок #. Если он стоит - уберем его. Таким образом, всем файлам с расширением .php будет поставлен в соответствие MIME-тип application/x-httpd-php.**

**Типы MIME используются для идентификации типа содержимого. Благодаря этому приложения смогут определять, какого вида данные присланы и в каком соответствии сети проводить их обработку. Перечень всех типов MIME, поддерживаемых системой, находится в файле C:\Program Files\Apache Group\Apache\conf. Тип MIME указывается в формате: тип/подтип. Вот основные типы MIME:**

**application - приложение (программа или документ, связанный с программой).**

**audio - аудиофайл (audio/mpeg).**

**image - графический файл (audio/png).**

**text - текст (text/html - в формате HTML, text/plain - простой).**

**MIME-тип пересылаемых по сети данных указывается в заголовке Content-type.**

**Сразу же после этой строки добавим строки:**

```
ScriptAlias/_php_"c:/Program Files/PHP5/" Action application/x-  
httpdhp "/_php_/php.exe"
```

**В этих двух строках мы сначала создаем псевдоним `_php_` для каталога `c:/Program Files/PHP 4/`, а затем связываем все файлы типа `application/x-httpd-php` с обработчиком `php.exe`.**

**Теперь можно протестировать PHP. Для этого создадим файл `test.php`:**

```
<?  
phpinfo ();? >
```

**Данный файл нужно поместить в каталог `c:\www\html`. В браузере введите `http://localhost/test.php`. В итоге вы должны увидеть результат работы функции `phpinfo()`.**

## **1.8 УСТАНОВКА MYSQL**

**Сервер MySQL можно загрузить по адресу `http://www.mysql.com`. Программа установки установит сервер в каталог `C:\Program Files\MySQL\`. Для запуска MySQL создадим файл `server.bat`:**

```
@echo off  
"C:\Program Files\MySQL\bin\mysqld"  
start/m "C:\Program Files\Apache Group\Apache\apache"
```

**Если вы работаете в Windows NT или 2000, `server.bat` должен выглядеть так, как это показано в следующем листинге:**

**@echo off**

**start C: \Progra~1\MySQL\bin\mysqld-nt - standalone**

**C: \Progra~1\Apache~1\Apache\apache - k start**

Теперь для запуска Apache и MySQL нужно ввести команду server. Создадим файл shutdown. bat, который будет останавливать оба сервера:

**@echo off**

**C: \Progra~1\Apache~1\Apache\apache - k shutdown**

**C: \Progra~1\MySQL\bin\mysqladmin - u root shutdown**

Также необходимо добавить в конец файла my. ini следующие строки (конкретно для данного проекта), задающие логин и пароль доступа к серверу:

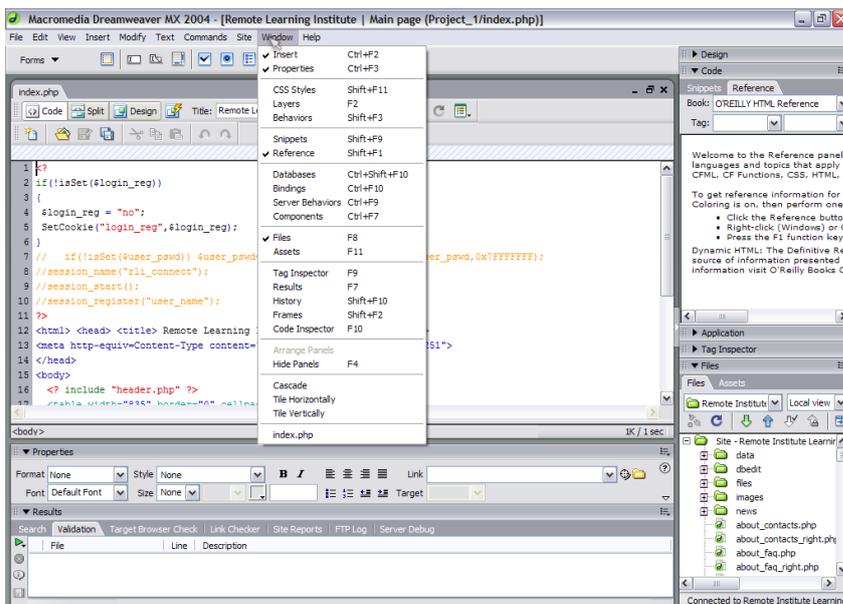
**user=root**

**password=geijr**

Файл настройки сервера баз данных MySQL находится по умолчанию в корневом каталоге Windows. Просмотреть все базы данных и таблицы, созданные на сервере MySQL можно утилитой WinMySQLAdmin 1.4

## 1.9 РАБОТА В MACROMEDIA DREAMWEAVER MX 2013

Интерфейс пакета Dreamweaver показан на рисунке 2.5 Помимо строки заголовка, строки меню, панели инструментов и стандартного рабочего окна программы отметим наличие нескольких палитр, которые, как и в графических редакторах, размещаются в рабочем окне программы.



**Рисунок 2.5 - Интерфейс пакета Dreamweaver MX 2004 с командами меню Window.**

**Остановимся на палитре Object (Объекты), внешний вид которой представлен на рисунке 2.6 Внешний вид палитры изменяется в зависимости от типа объекта, выбранного в раскрывающемся списке, расположенного в ее верхней части; таким образом палитру можно назвать адаптивной. По умолчанию в списке устанавливается раздел Common (Основные). На рисунке 2.6 выбран именно этот вариант.**

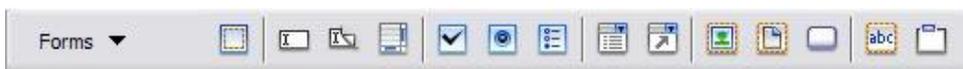


**Рисунок 2.6 - Палитра Objects раздел Common**

**Эта палитра с помощью различных инструментов позволяет включать в состав страницы различные элементы, в частности, по порядку расположения кнопок в палитре в**

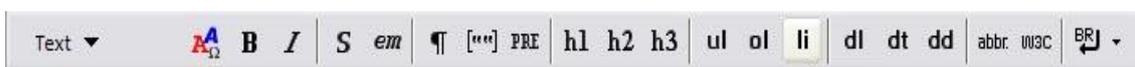
направлении слева направо и сверху вниз: **Image** (Изображение), **Rollover Image** (Анимированную кнопку), внешний вид которой изменяется при подведении к ней курсора мыши, **Table** (Таблицу), **Tabular Data** (таблицу с занесенными в нее данными, подготовленными заранее в другом программном пакете), **Draw Layer** (Новый изобразительный слой), **Navigation Bar** (Навигационную панель со специальными кнопками для перемещения по сайту), **Horizontal Rule** (Горизонтальную линейку), **E-mail Link** (Ссылку на электронный адрес), **Date** (Текущую дату), **Server-Side Include** (Активизацию серверной стороны), **Flash Button** (Анимированную Flash-кнопку), **Flash Text** (Анимированный Flash-текст), **Shockwave** (Анимационный файл в Shockwave-формате), **Generator** (средства, делающие Web-страницы динамическими, в частности, апплеты и элементы ActiveX, объекты, воспроизводимые с помощью дополнительных модулей) и пр.

Палитра **Objects** раздел **Forms** (Формы), показанная на рисунке 2.7, служит для создания форм и содержит следующие кнопки: **Form** (Форма), **Text field** (текстовое поле), **Button** (Кнопка), **Checkbox** и **Radio Button** (Переключатели), **List/menu** (Список/меню), **File field** (Поле для передачи файлов) и **Image field** (Поле для передачи изображений), **Hidden field** (Скрытое поле) и **Jump menu** (Указатель гиперссылки для перехода на другой сайт).



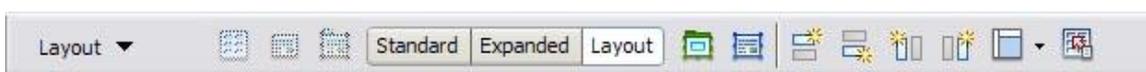
**Рисунок 2.7 - Палитра Objects раздел Forms**

Палитра Objects раздел Text (Формы), показанная на рисунке 2.8 позволяет включать в страницу различные элементы текстового форматирования - жирный текст, наклонный, создавать списки, переносы и т.п.



**Рисунок 2.8 - Палитра Objects раздел Text**

В нижней части всех палитр группы Object (Объект) предусмотрены поля Layout (Выравнивание) и View (Просмотр), которые позволяют наблюдать и управлять размещением объектов на Web-странице (рисунок 2.9).



**Рисунок 2.9 - Палитра Objects раздел Layout**

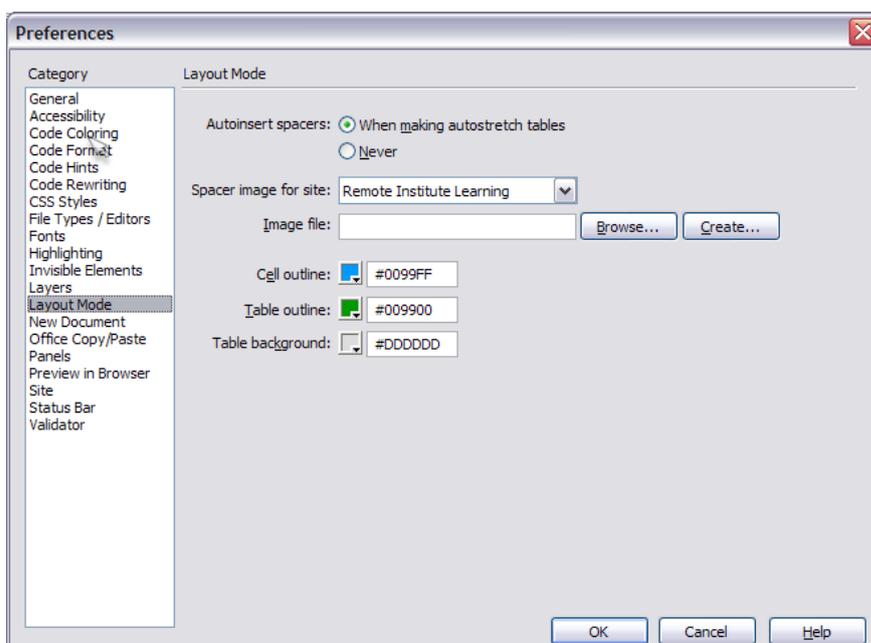
И наконец, важная палитра, облегчающая во многом работу со скриптами PHP - палитра Object - PHP. Предусмотрена вставка различных элементов языка PHP, например открывающего и закрывающего тэгов, операторов и т.п. (рисунок 2.10).



**Рисунок 2.10- Палитра Objects раздел PHP**

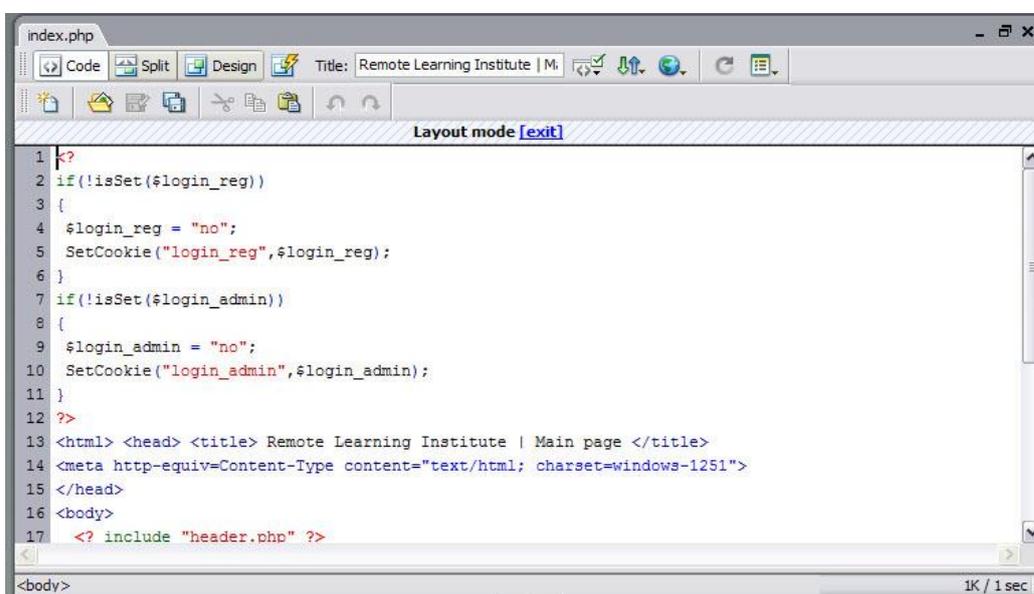
**Прежде всего, надо тщательно продумать общую структуру сайта, а также структуру каталогов (папок) и размещение в них файлов. В начале работы следует разместить в корневом каталоге сайта все PHP-файлы и несколько подкаталогов, в том числе: изображений, создания баз данных, файлов, данных, новостей.**

**Для непосредственного создания сайта следует предварительно сделать все необходимые установки. С этой целью выбирается команда Edit | Preferences (Редактирование | Настройки). В результате откроется диалоговое окно Preferences (Настройки), показанное на рисунке 2.11 Здесь задаются различные настройки для будущего сайта - название, цвета ссылок, размещение каскадных таблиц стилей CSS, параметры использования различных элементов.**



**Рисунок 2.11 - Диалоговое окно Preferences**

Для изменения любой страницы в Dreamweaver используется редактор страниц. На рисунке 2.12 показана рабочая область с загруженным документом `inde.php`. Редактировать можно в трех режимах - кода, дизайнера, совмещенного (кода и дизайнера). Для работы с PHP больше всего подходит первый режим. Просматривать страницы и запускать скрипты удобно по клавише F12, которая автоматически открывает страницу в выбранном заранее браузере.



```
index.php
Code Split Design Title: Remote Learning Institute | M
Layout mode [exit]
1 <?
2 if(!isset($login_reg))
3 {
4     $login_reg = "no";
5     SetCookie("login_reg",$login_reg);
6 }
7 if(!isset($login_admin))
8 {
9     $login_admin = "no";
10    SetCookie("login_admin",$login_admin);
11 }
12 ?>
13 <html> <head> <title> Remote Learning Institute | Main page </title>
14 <meta http-equiv=Content-Type content="text/html; charset=windows-1251">
15 </head>
16 <body>
17 <? include "header.php" ?>
</body>
1K / 1 sec
```

Рисунок 2.12 - Редактор кода страниц

Для синхронизации изменений внесенных при модификации страниц сайта или добавлении данных (файлов или рисунков) используется редактор файлов (рисунок 2.13). Добавлять или удалять файлы можно из контекстного меню редактора, выбирая соответствующие пункты. Удобнее всего редактировать данные и модифицировать страницы в локальном режиме Local View, а затем синхронизировать с сервером в режиме Testing Server через пункт меню Synchronize.



**Рисунок 2.13 - Редактор файлов сайта**

## II ГЛАВА ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ ИНФОРМАЦИОННО-ДОПРИЗЫВНОЙ СИСТЕМЫ

### 2.1 РАЗРАБОТКА БАЗЫ ДАННЫХ

Одним из значимых элементов любого сайта является пользовательский интерфейс. Удобство и простота его реализации во многом определяют успех любого проекта. Неудачная подборка цветовой гаммы, в том числе несочетающихся цветов, бывает причиной нежелания пользователей посещать даже насыщенный полезной информацией веб-ресурс. Для облегчения вывода текста созданы каскадные таблицы стилей [14]. А в качестве языка программирования используется PHP, эффективно взаимодействующий с базами данных и позволяющий реализовывать серьезные проекты.

### 2.2 Разработка пользовательского интерфейса

В качестве цветовой гаммы выбраны синие цвета и оттенки голубых, достаточно хорошо гармонирующие друг с другом. Текст отображен шрифтом семейства Verdana, Arial, Sans-serif черным, темно синим или красным цветом (Только в заголовочной части).

Основными элементами интерфейса являются: верхняя заголовочная часть, содержащая меню, левая информационная панель, рабочая область страницы, часть гостевой книги и нижняя часть, содержащая ссылку на страницу администрирования сайта.

### 2.3 ФАЙЛОВАЯ СТРУКТУРА БАЗЫ ДАННЫХ

**Все файлы приложения размещаются в корневом каталоге сайта. Проект содержит несколько подкаталогов:**

**InternetTechnology, хранит файлы подразделов дисциплин;**

**images, хранит все файлы изображений, используемые в приложении;**

**smiles, хранит смайлики для чата и гостевой книги.**

**В корневом каталоге хранятся файлы, описанные в таблице 2.13.**

**Таблица 2.13- Описание файловой структуры приложения**

| № п/п | Название файла               | Назначение файла   |
|-------|------------------------------|--|
| 1     | Admchat. php                 | Страница администрирования чата  |
| 2     | Admexam. php                 | Страница администрирования экзаменатора. Удаление предметов или вопросов   |
| 3     | Adminguest. php              | Страница администрирования гостевой книги  |
| 4     | AdminInternetTechnology. php | Главная страница администратора. Связана со страницами администрирования: admchat. php, admexam. php, adminguest. php, input. php, students. php |
| 5     | Adminpass. php               | Вход на страницу администратора  |
| 6     | All. php                     | Файл связи со всеми файлами функций  |
| 7     | BaseFunctions. php           | Основные функции   |
| 8     | Chat. php                    | Страница чата  |
| 9     | Exam. php                    | Страница тестирования  |
| 10    | FuncChat. php                | Функции для страницы чата  |
| 11    | FuncExam. php                | Функции для страницы тестирования  |
| 12    | FuncGuest. php               | Функции для гостевой книги   |
| 13    | Gb. php                      | Модуль гостевой книги  |
| 14    | Input. php                   | Страница ввода новых дисциплин и вопросов. Доступна только администратору  |
| 15    | InternetTechnology. php      | Главная страница модуля Интернет технологии  |
| 16    | LeftAdmin. php               | Левая часть страницы администрирования   |
| 17    | Login. php                   | Модуль входа в систему по Никю и паролю  |
| 18    | Postmodule. php              | Модуль, использующийся в гостевой книге и поиск для посылки сообщений  |
| 19    | RegisterUser. php            | Модуль первоначальной регистрации  |
| 20    | Rtf1. php                    | Модуль производящий сертификат   |
| 21    | Rtfs. php                    | Модуль производящий список студентов прошедших тестирование  |
| 22    | Score. php                   | Станица результатов тестирования   |
| 23    | StartSession. php            | Модуль обработки результатов регистрации   |
| 24    | Store. php                   | Модуль записи вопросов тестирования  |
| 25    | Students. php                | Страница результатов тестирования. Доступна только администратору  |
| 26    | Shop. php                    | Модуль интернет - магазина   |
| 27    | Shopping. php                | Модуль регистрации заказчика   |
| 28    | Rtfshop. php                 | Модуль печати заказов в Word   |
| 29    | Admshop. php                 | Администрирование интернет - магазина  |
| 30    | Funcshop. php                | Функции интернет - магазина  |
| 31    | Styles. css                  | Подключаемый модуль таблицы стилей   |

## 2.4 ВХОД В СИСТЕМУ MYSQL

Для входа в систему MySQL в интерфейс командной строки на своей машине и наберите следующее:

**> mysql. exe - h localhost - u root**

Командная подсказка может выглядеть по-разному, в зависимости от используемых операционной системы и командной оболочки.

Команда `mysql` запускает монитор MySQL. Это клиент командной строки, который выполняет соединение с сервером MySQL.

Ключ `- h` используется для обозначения хоста, к которому осуществляется подключение - это компьютер с запущенным

**сервером MySQL. При вводе этой команды на той же машине, на которой находится сервер MySQL, применять этот ключ, равно как и параметр hostname, вовсе не обязательно. Если на другой, то параметр hostname следует заменить именем машины, на которой выполняется сервер MySQL.**

**Ключ - u используется для указания имени пользователя, под которым требуется осуществить соединение.**

**(Если этого не произошло, убедитесь в том, что MySQL-сервер запущен и где-то в. вашем пути присутствует команда mysql)**

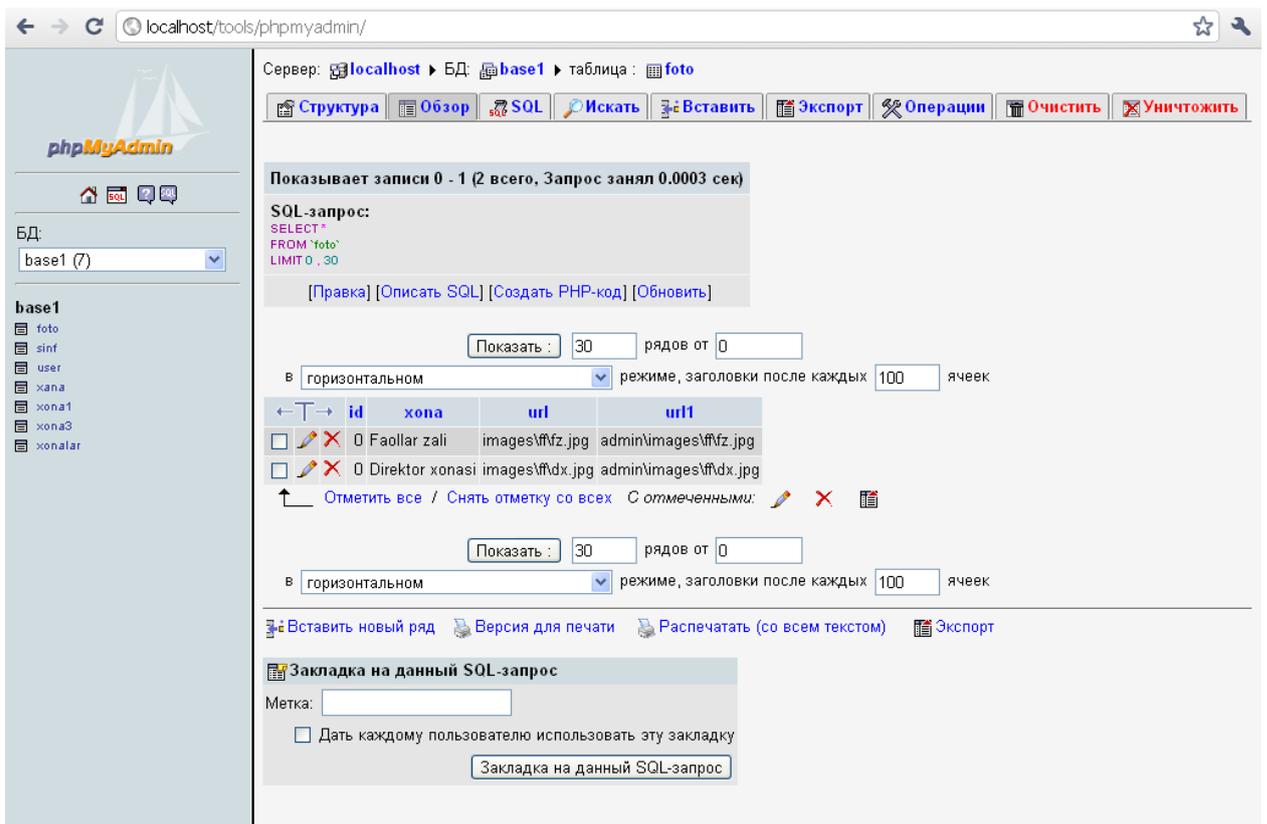
**Необходимо ввести пароль.**

## **2.5 СОЗДАНИЕ ТАБЛИЦ БАЗЫ ДАННЫХ**

**Система баз данных MySQL может поддерживать множество различных баз данных. Обычно на одно приложение будет существовать одна база данных.**

**Для создания базы данных (itn) используют подключаемый файл “create\_itn. sql”. Для этого нужно ввести команду:**

**Mysql>\. create\_itn. sql f: \create\_itn. sql**



как показано на рисунке 2.14

```
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 62 to server version: 4.0.21-nt
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
mysql> \.create_itn.sql f:\create_itn.sql;
```

Рисунок 2.14 - создание таблиц с помощью sql-файла

где f: \create\_itn. sql - имя и путь к файлу.

Текст файла - “create\_itn. sql “ приведен ниже.

**create database itn; // Создать базу данных под названием “itn”**

**use itn; // использовать базу данных ”itn”**

**create table chatmessages // Создать таблицу “chatmessages”**

**( // Сообщения чата**

**idchmes int unsigned not null auto\_increment primary key,**

**idstudc int not null, // id студента**

**message char(150) not null, // сообщение чата**

**ip char(15), // IP - адрес**

```

time datetime not null // дата записи сообщения
);
create table guests // Создать таблицу "guests"
( // гостевая книга
idguest int unsigned not null auto_increment primary key,
idstudg int not null, // id студента
message char(150) not null, // сообщение
ip char(15), // IP - адрес
page char(30) not null, // страница гостевой книги
time datetime not null // дата записи
);
create table subjects // Создать таблицу "subjects"
( // Предметы для тестирования
idsub int unsigned not null auto_increment primary key,
subject char(30) not null // предмет
);
create table questions // Создать таблицу "questions"
( // вопросы для тестирования
idques int unsigned not null auto_increment primary key,
idsubq int not null, // родительский предмет
quesmes char(150) not null // вопрос
);
create table answers // Создать таблицу "answers"
( // ответы для тестирования
idans int unsigned not null auto_increment primary key,
parent int not null, // id - вопроса

```

```

ansmes char(150) not null, // ответ
rightanswer int default 0 not null // правильный ответ = 1
);
create table students // Создать таблицу “students”
( // Студенты прошедшие тест
idstud int not null primary key, // шифр
name char(20) not null, // имя
secondname char(20) not null, // фамилия
faculty char(20) not null, // факультет
email char(20) not null, // e-mail
login char(20) not null, // login входа в систему
password char(20) not null, // пароль
lastdate datetime not null // Дата последнего входа в систему
};
create table results
{
idres int not null auto_increment primary key,
idstudr int not null, // id - студента
idsubr int not null, // id - предмета
posted datetime not null, // дата сдачи
score int not null, // результат
count tinyint(4) not null, // IP - количество попыток
);
create table ips // Создать таблицу “ips”
( // запрещенных IP - адресов
idip int unsigned not null auto_increment primary key,

```

```

ip char(15) not null, // IP - адрес
sub char(20) not null // станица(гостевая книга или чат)
);
create table bals // Создать таблицу "bals"
( // баллы для тестирования
bal int not null, // бал
percent int not null // проценты
);
create table cd
(
idc int unsigned not null auto_increment primary key,
title char(50) not null,
description char(150),
price float not null
);
create table zakaz
(
idz int unsigned not null auto_increment primary key,
idclinta int not null,
idcd int not null,
count tinyint default 0 not null,
recdate datetime not null,
flag_done enum('y', 'n') DEFAULT 'n' NOT NULL
);
create table shopclients
(

```

```

idclient int unsigned not null auto_increment primary key,
name char(10) not null,
sname char(15) not null,
email char(30) not null,
address text not null,
fhone char(15),
lastdate datetime not null
);
grant select, insert, update, delete // определение прав
пользователя
on itn.*
to itn@localhost identified by 'password';

```

**Таблица 2.14 - Используемые типы данных**

| п/п | Тип данных                  | Описание  |
|-----|-----------------------------|---|
|     | int                         | Целое число. В MySQL тип данных int может быть со знаком или без знака  |
|     | tinyint                     | Очень маленькие целые числа   |
|     | char(длина)                 | Символьная величина фиксированной длины. Поля типа char не могут содержать строки длины большей, чем указанное значение. Поля меньшей длины дополняются пробелами |
|     | date                        | Стандартное значение даты. Формат хранения даты по умолчанию 'уууу-mm-dd'   |
|     | text                        | Символьная величина переменной длины  |
|     | enum('знач1', ..., 'значN') | Перечисляемый тип   |

### **Что означают другие ключевые слова**

**NOT NULL** означает, что все строки таблицы должны иметь значение в этом атрибуте. Если **NOT NULL** не указано, поле может быть пустым (**NULL**).

**AUTO\_INCREMENT** - специальная возможность MySQL, которую можно задействовать в числовых столбцах. Если при вставке строк в таблицу оставлять такое поле пустым, MySQL автоматически генерирует уникальное значение идентификатора. Это значение будет на единицу больше максимального значения, уже существующего в столбце. В каждой таблице может быть не больше одного такого поля. Столбцы с **AUTO\_INCREMENT** должны быть проиндексированными.

**PRIMARY KEY** после имени столбца определяет, что этот столбец является первичным ключом для таблицы. Данные в этом столбце должны быть уникальными. MySQL автоматически индексирует этот столбец. Автоматический индекс по первичному ключу сберегает индекс, требуемый **AUTO\_INCREMENT**.

Указывать **PRIMARY KEY** после названия столбца следует лишь тогда, когда мы имеем дело с первичным ключом в виде одиночного столбца. Альтернативным вариантом является конструкция **PRIMARY KEY** в конце описания таблицы `order_items`. В последнем случае первичный ключ включает два столбца.

**UNSIGNED** после целочисленного типа означает, что его значение может быть либо положительным, либо нулевым.

На рисунке 2.15 показан результат обработки подключаемого файла SQL.

```
Database changed
Query OK, 0 rows affected (0.11 sec)
Query OK, 0 rows affected (0.05 sec)
Query OK, 0 rows affected (0.06 sec)
Query OK, 0 rows affected (0.08 sec)
Query OK, 0 rows affected (0.05 sec)
Query OK, 0 rows affected (0.06 sec)
Query OK, 0 rows affected (0.05 sec)
Query OK, 0 rows affected (0.06 sec)
Query OK, 0 rows affected (0.05 sec)
Query OK, 0 rows affected (0.01 sec)
mysql> _
```

Рисунок 2.15 - создание таблиц после обработки файла SQL

Для просмотра созданных таблиц в базе данных нужно ввести команду **SHOW TABLES** (рисунок 2.16):

**Mysql>Show tables;**

```
mysql> use itn;
Database changed
mysql> show tables;
+-----+
| Tables_in_itn |
+-----+
| answers       |
| bals          |
| chatmessages  |
| guests        |
| ips           |
| questions     |
| results       |
| students      |
| subjects      |
+-----+
9 rows in set (0.00 sec)
mysql> _
```

Рисунок 2.16 - просмотр созданных таблиц

Команда **DESCRIBE** дает возможность увидеть дополнительную информацию по конкретной таблице (рисунок 2.17)

```
mysql> describe students;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default      | Extra |
+-----+-----+-----+-----+-----+-----+
| idstud     | int(11)   |      | PRI | 0             |       |
| name       | char(20)  |      |     |               |       |
| secondname | char(20)  |      |     |               |       |
| faculty    | char(20)  |      |     |               |       |
| email      | char(30)  |      |     |               |       |
| login      | char(10)  |      |     |               |       |
| password   | char(10)  |      |     |               |       |
| lastdate   | date      |      |     | 0000-00-00   |       |
+-----+-----+-----+-----+-----+-----+
8 rows in set (0.00 sec)

mysql>
```

**Рисунок 2.17 - просмотр типов полей таблицы**

Эти команды полезны, если требуется вспомнить, какие типы столбцов используются.

## 2.6 ФОРМИРОВАНИЕ СТРАНИЦ БАЗЫ ДАННЫХ

Рассмотрим создание страниц приложения на примере главной страницы модуля сайта "Интернет технологии". Скелет страницы состоит из заголовочной части, таблицы из двух столбцов, в левой части которой располагается форма регистрации и кнопка входа на страницу тестирования и кнопка входа в чат, в правой отображается основное содержимое страницы, гостевой книги, находящейся сразу под основной таблицей и нижней части со ссылкой на страницу входа в администрирования. (Рисунок 2.18)

|  |
|--|
| <b>Заголовочная часть</b>                    |
| <b>Часть кнопок навигации по дисциплинам</b> |

|   |  |
|---|--|
| <p><b>Часть<br/>формой<br/>входа<br/>систему<br/>кнопками<br/>входа<br/>тестирование<br/>и входа<br/>чат.</b></p> | <p><b>с<br/>Основная часть<br/>в<br/>и<br/>в<br/>в</b></p> |
| <p><b>Часть гостевой книги</b></p>  |  |
| <p><b>Нижняя часть</b></p>  |  |

**Рисунок 2.18 - скелет главной страницы**

Все скрипты PHP заключаются в скобки `<? и? >`. Язык позволяет формировать страницы из нескольких отдельных составляющих. Для этого используются функции `include` и `require`. Например, скрипт

```
<? include "all. php"? >
```

Делает доступными для страницы все функции подключенные в этом файле.

```
<? doheader('Интернет технологии');? >
```

Включает в страницу функцию `doheader()`, отображающую заголовочную часть с логотипом и названием сайта. (Рисунок 2.19)

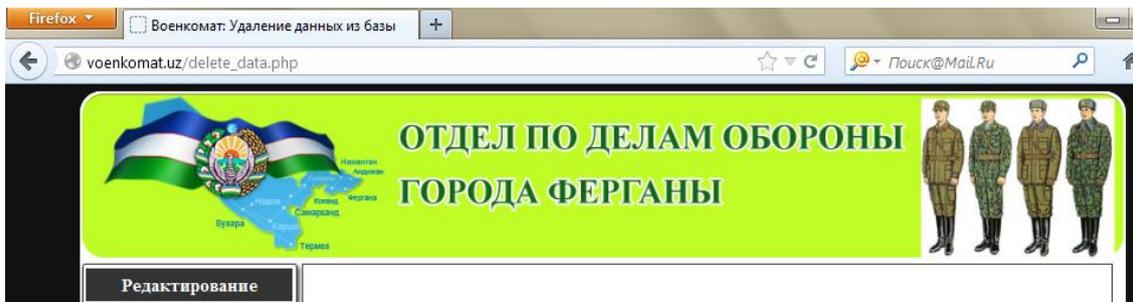


Рисунок 2. 19 - заголовочная часть

Следующая часть кода выводит кнопки навигации по дисциплинам, отображенным на рисунке 2. 20:

```
<? do_buttons()? >
```

```
function do_buttons() // Кнопки
```

```
{
```

```
? >
```

```
<center><table width="100%" cellpadding=6>
```

```
<td class="bor" align="center" width="25%">
```

```
<a href="InternetTechnology. php? mode=enter">
```

```
<font size="4" color="#0000FF">Введение</font></a></td>
```

```
<td class="bor" align="center" width="25%">
```

```
<a href="InternetTechnology. php? mode=lections">
```

```
<font size="4" color="#0000FF">и</font></a></td>
```

```
<td class="bor" align="center" width="25%">
```

```
<a href="InternetTechnology. php? mode=labs">
```

```
<font size="4" color="#0000FF"> </font></a></td>
```

```
<td class="bor" align="center" width="25%">
```

```
<a href="InternetTechnology. php? mode=ind">
```

```
<font size="4" color="#0000FF"> </font></a></td>
```

```
</table></center>
```

```
<?
```

}, где **MODE** указывает основной части таблицы на то, что должно выводиться.

The image shows a web interface with two main sections. On the left, there is a sidebar with a dark header 'Редактирование' (Editing) containing four buttons: 'Главная »' (Home), 'Добавить »' (Add), 'Изменить »' (Edit), and 'Удалить »' (Delete). Below this is another dark header 'Призывники' (Recruits) with a list of names: '» Рогов Виктор Николаевич', '» Тураходжаев Ильяс Нажмиддинович', and '» Холматов Ислон Улугбекович'. On the right, there is a registration form titled 'Учетная карточка 1' (Registration Card 1) in red. Below the title is a red instruction: 'Ниже введите личные данные призыв поля)' (Below enter personal data of the recruit field). The form contains several input fields: 'Учетный номер призыва' (Recruitment number), 'Фамилия:' (Surname), 'Имя:' (Name), 'Отчество:' (Patronymic), 'Серия паспорта:' (Passport series), and 'Номер паспорта:' (Passport number).

Рисунок 2.20 - кнопки навигации по дисциплинам

Затем в левой части таблицы выводится форма регистрации и кнопки тестирования и чата. (Рисунок 2.21)

<?

```

echo"<form action=login. php method=post>";
echo"<br><table width=155 bgcolor=$color1 border=2
bordercolor=darkblue cellpadding=3><tr><td><font
color=blue><b>Войти в систему</b></font><hr color=#ff9900>
Ник<br><input type=text name=login><br>
Пароль</font><br><input type=password
name=password><br><input type=submit name=submit
value='Вход'><br>
<a href=InternetTechnology. php?

```

```

mode=register>Регистрация</a></td></tr></table>";
echo "</form>";
echo"<br><table width=155 bgcolor=blue border=2
bordercolor=darkblue><tr><td align=center><a href=exam.
php><font color=white
size=+1>Тестирование</font></a></td></tr></table>";
echo"<br><table width=155 bgcolor=blue border=2
bordercolor=darkblue><tr><td align=center><a href=chat.
php><font color=white
size=+1>Общение</font></a></td></tr></table><br>"
? >

```

The image shows a web form with three input elements. The first is a text input labeled 'Логин' (Login) containing the text 'Login...'. The second is a password input labeled 'Парол' (Password) filled with ten black dots. The third is a button labeled 'Кириш' (Login/Submit).

**Рисунок 2.21 - левая часть страницы с формой входа и кнопками**

Здесь `echo"<form action=login. php method=post>";` означает что данные формы входа будут посланы модулю проверки данных `login. php`. Метод `POST` отличается от метода `GET` тем, что при использовании метода `GET` значения полей присоединяются к URL, указанному в атрибуте `ACTION`.

Далее в правой части выводится основная часть страницы, выбранная пользователем с помощью кнопок навигации. (С помощью переменной `MODE`).

```
switch($mode)
{
case 'register': // подключить модуль регистрации
include('register_user.php');
break;
case 'enter': // Модуль введения
$pagegb='Интернет_технологии'; // Эта переменная указывает
гостевой книге // индентификатор страницы.
include('InternetTechnology\enter.php');
break;
case 'lections': // Подключить модуль $pagegb='';
include('InternetTechnology\lections.php');
break;
case 'labs':
$pagegb=''; //
include('InternetTechnology\labs.php');
break;
case 'ind': ///
$pagegb='';
include('InternetTechnology\ind.php');
break;
default:
```

```
include('InternetTechnology\enter. php'); // по умолчанию  
подключить модуль введение  
break;  
}
```

**Модули, подключаемые в этой части имеют вид:**

```
<?
```

**Текстовый материал с использованием тегов логического и физического форматирования текста.**

```
? >
```

**Затем, ниже основной таблицы, выводится гостевая книга, относящаяся к данной странице. (Рисунок 2.22)**

```
<? include('gb. php');? >
```

**Просто подключается модуль гостевой книги 'gb. php', который в свою очередь включает модуль отправки сообщений: 'postmodule. php', который также используется на странице чата.**

**Ниже вызовом функции doout(), выводится нижняя часть изображения со ссылкой на страницу входа в администрирование. (Рисунок 2.23)**



**Рисунок 2.23 - нижняя часть страницы**

**Общий вид главной страницы модуля , базы данных по дисциплине "Интернет технологии" со значением переменной MODE равной lectures ()**

## 2.7 ИСПОЛЬЗОВАНИЕ ПЕРЕМЕННЫХ СЕАНСА

**HTTP** иногда называют "протоколом без состояния". Это означает, что данный протокол не имеет встроенного способа поддержки состояния между двумя транзакциями. Когда пользователь запрашивает друг за другом две страницы, **HTTP** не обеспечивает возможности уведомить, что оба запроса исходят от одного и того же пользователя.

Таким образом, идея управления сеансами заключается в обеспечении отслеживания пользователя в течение одного сеанса связи с **Web**-сайтом.

Если это удастся осуществить, мы сможем легко поддерживать подключение пользователя и предоставление ему содержимого сайта в соответствии с его уровнем прав доступа или персональными настройками. Мы сумеем отслеживать поведение пользователя.

Для запуска сеанса в **RHP** используется уникальный идентификатор сеанса, представляющий собой зашифрованное случайное число. Идентификатор сеанса генерируется **RHP** и сохраняется на стороне клиента в течение всего времени жизни сеанса. Для хранения идентификатора сеанса используется либо **cookie**-набор на компьютере пользователя, либо **URL**.

Идентификатор сеанса играет роль ключа, обеспечивающего возможность регистрации некоторых специфических переменных в качестве так называемых переменных сеанса. Содержимое этих переменных сохраняется на сервере. Единственной информацией, "видимой" на стороне клиента,

**является идентификатор сеанса. Если во время определенного подключения к вашему сайту идентификатор сеанса является "видимым" либо в cookie-наборе, либо в URL, имеется возможность получить доступ к переменным сеанса, которые сохранены на сервере для данного сеанса. По умолчанию переменные сеанса хранятся в двумерных файлах на сервере (при желании способ хранения можно изменить и использовать вместо двумерного файла базу данных, но для этого потребуется написать собственную функцию).**

**Скорее всего, придется иметь дело с Web-сайтами, на которых для хранения идентификатора сеанса используется URL. Если в вашем URL имеется строка данных, которые выглядят случайными, то это, скорее всего, свидетельствует об использовании одной из двух описанных здесь разновидностей управления сеансом.**

**Другим решением проблемы сохранения состояния на протяжении некоторого количества транзакций, при наличии чистого внешнего вида URL, являются cookie-наборы.**

## **2.8 ИСПОЛЬЗОВАНИЕ COOKIE - НАБОРОВ В СЕАНСЕ**

**При использовании cookie-наборов возникают некоторые проблемы: есть браузеры, которые не принимают cookie-наборы, а есть пользователи, которые запрещают использование cookie-наборов в своих браузерах. Это одна из причин, по которым в сеансах PHP используются двойной**

метод cookie-набор/адрес URL (ниже этот вопрос рассматривается более подробно).

В сеансе PHP нет необходимости задавать cookie-наборы вручную. Это за вас сделают функции сеанса.

Для того чтобы просмотреть содержимое cookie-набора, установленное при управлении сеансом, можно воспользоваться функцией `session_get_cookie_params()`. Она возвращает ассоциативный массив, содержащий элементы `lifetime`, `path` и `domain`.

Можно использовать также:

```
session_set_cookie_params($lifetime, $path, $domain);
```

этот оператор устанавливает параметры cookie-набора для сеанса.

### 2.8.1 РЕАЛИЗАЦИЯ УПРАВЛЕНИЯ ПРОСТЫМ СЕАНСОМ

Основными этапами использования сеанса являются следующие: Запуск сеанса;

Регистрация переменных сеанса;

Использование переменных сеанса;

Отмена регистрации переменных и закрытие сеанса.

Заметим, что все перечисленные этапы не обязательно могут содержаться в одном сценарии, и некоторые из них могут находиться в нескольких сценариях. Рассмотрим каждый из этих этапов последовательно.

## 2.8.2 ЗАПУСК СЕАНСА

Прежде чем можно будет воспользоваться функциональными возможностями сеанса, следует запустить сам сеанс. Существует три способа сделать это.

Первый (и самый простой) заключается в том, что сценарий начинается с вызова функции `session_start()`.

Эта функция проверяет, существует ли идентификатор текущего сеанса. Если нет, она его создает. Если же идентификатор текущего сеанса уже существует, она загружает зарегистрированные переменные сеанса, чтобы они стали доступными для использования.

Надо отметить, что это прекрасный способ - вызов `session_start()` в начале сценариев, в которых используются сеансы.

Второй способ заключается в том, что сеанс запускается при попытке зарегистрировать переменные сеанса.

Третий способ запустить сеанс - задать установки PHP, при которых сеанс будет запускаться автоматически, как только кто-либо посетит ваш сайт. Для этого следует воспользоваться опцией `session.auto_start` в файле `php.ini`.

## 2.8.3 РЕГИСТРАЦИЯ ПЕРЕМЕННЫХ СЕАНСА

Для того чтобы получить возможность отслеживать переменные от одного сценария к другому, их необходимо зарегистрировать. Это делается путем вызова функции `session_register()`.

Данный оператор регистрирует имя переменной и отслеживает ее значение. Отслеживание переменной будет осуществляться, пока не завершится сеанс либо пока вручную не отменится ее регистрация.

```
If(! session_id()) session_start();
```

```
If(! isset($_session ['error_fill']) $_session ['error_fill'] ['bool'] =false
```

Здесь регистрируется переменная сеанса “error fill”, используемая для отображения ошибки заполнения полей регистрации студента.

#### 2.8.4 ИСПОЛЬЗОВАНИЕ ПЕРЕМЕННЫХ СЕАНСА

Чтобы сделать переменную сеанса доступной для использования, сначала необходимо запустить сеанс, воспользовавшись одним из описанных выше способов.

После этого появляется доступ к этой переменной. Если опция `register_globals` включена, то доступ к этой переменной можно получить через сокращенную форму ее имени, например, `$error_fill`. Если же упомянутая опция не включена, получить доступ к переменной можно через ассоциативный массив `$_SESSION`, например, `$_SESSION ['error_fill']`.

Переменные сеанса не могут быть перезаписаны данными GET или POST. Это хорошо с точки зрения обеспечения безопасности, однако сопряжено с некоторыми ограничениями при кодировании.

С другой стороны, от вас потребуется тщательность при проверке на предмет того, установлены ли уже переменные

сеанса (например, с использованием `isset()` либо `empty()`). Кроме того, следует иметь в виду, что переменные могут быть установлены пользователем через GET или POST. Проверить, является ли переменная зарегистрированной переменной сеанса, можно обратившись к функции `$_session ()`. Вызов функции выполняется следующим образом:

```
$_session ['error_fill']
```

Следующая функция проверит правильность написания электронного адреса и вернет, `$_session ['error_fill'] [bool] =true`, если адрес написан неправильно.

```
If(! ereg(^. +@. +\|. +$, $email))
```

```
{
```

```
$session ['error_fill'] [bool] =true;
```

```
$_session ['error_fill'] ['message'] ="Проверьте правильность написания электронного адреса";
```

```
}
```

Если переменная `$_session ['student']` существует то ей присваиваются текущие значения. Здесь переменной присваиваются значения регистрации студента.

```
$_SESSION ['student'] ['name'] =$name; $_SESSION ['student'] ['family'] =$family;
```

```
$_SESSION ['student'] ['group'] =$group; $_SESSION ['student'] ['faculty'] =$faculty;
```

```
$_SESSION ['student'] ['email'] =$email; $_SESSION ['student'] ['login'] =$login;
```

## 2.8.5 ОТМЕНА РЕГИСТРАЦИИ ПЕРЕМЕННЫХ И ЗАВЕРШЕНИЕ СЕАНСА

После окончания работы с переменной сеанса ее регистрацию можно отменить, воспользовавшись функцией `session_unregister()`:

```
$_session_unregister('error_fill')
```

Подобно функции регистрации, эта функция требует указания имени переменной, регистрацию которой необходимо отменить, в виде строки, не включающей символ `$`. Данная функция за один раз может отменить регистрацию только одной переменной сеанса (в противоположность `session_register()`). Однако, для отмены регистрации всех переменных текущего сеанса можно обратиться к `session_unset()`.

По завершении сеанса сначала потребуется отменить регистрацию всех переменных, а затем вызвать `Session_destroy()`

для обнуления идентификатора сеанса.

## 2.8.6 ИСПОЛЬЗОВАНИЕ ПЕРЕМЕННЫХ СЕАНСА В ПРИЛОЖЕНИИ

При запуске приложения создаются 2 переменные сеанса: `$_SESSION ['mode']`, отвечающая за страницу приложения и `$_SESSION ['error_fill']`, отвечающая за правильность заполнения полей регистрации студентом.

```
if(! session_id()) session_start();  
if(! isset($_SESSION ['error_fill'])) $_SESSION ['error_fill'] ['bool']  
='false';  
if(isset($_REQUEST ['mode']))  
{
```

```

$mode=$_REQUEST ['mode'] ;
$_SESSION ['mode'] =$mode;
}
else $mode=isset($_SESSION ['mode'])? $_SESSION ['mode']: '';
$_SESSION ['client'].
$_SESSION ['client'] =isset($_SESSION ['client'])? $_SESSION
['client']: '';
if($_SESSION ['client'] =='')
{
if(isset($_SESSION ['student']))
{
$_SESSION ['client'] ['name'] =$_SESSION ['student'] ['name'] ;
$_SESSION ['client'] ['sname'] =$_SESSION ['student'] ['family'] ;
$_SESSION ['client'] ['email'] =$_SESSION ['student'] ['email'] ;
$_SESSION ['client'] ['fhone'] ='';
$_SESSION ['client'] ['address'] ='';
}else
{
$_SESSION ['client'] ['name'] ='';
$_SESSION ['client'] ['sname'] ='';
$_SESSION ['client'] ['fhone'] ='';
$_SESSION ['client'] ['email'] ='';
$_SESSION ['client'] ['address'] ='';
}

```

**Здесь сначала проверяется наличие переменной сеанса `$_SESSION ['student']`, и если она зарегистрирована,**

совпадающим полям переменной `$_SESSION` ['client'] присваиваются соответствующие поля переменной `$_SESSION` ['student'], иначе поля обнуляются.

## 2.9 ЗАПУСК БАЗЫ ДАННЫХ И ОПИСАНИЕ РАБОТЫ С НИМ

Для запуска базы данных необходимо запустить Интернет-браузер и набрать в строке адреса в локальном режиме

<http://localhost/kashtan/InternetTechnology.php>. Откроется

главная страница приложения (Рисунок 2.25). Пользователь получает возможность навигации по материалам сайта, регистрации (для незарегистрированных пользователей) или входа в систему (для зарегистрированных пользователей) и дистанционного обучения.

С левой стороны расположены поля для входа в систему, ссылка на форму регистрации и кнопки тестирования и чата.

## **III РАЗРАБОТКА АЛГОРИТМА И ОПИСАНИЕ РАБОТЫ ПРОГРАММЫ ОБРАБОТКИ ДАННЫХ**

### **3.1 ВЫБОР МОДЕЛИ ПРОЕКТИРОВАНИЯ**

**При разработке данного проекта упор был сделан на принципы модульного Web программирования, применены технологии Web – разработки – HTML, CSS Javascript, PHP. Для работы программы необходим Web – сервер Apache. Весь основной код программы написан на языке программирования PHP. Каждая составляющая часть программы вынесена в отдельный модуль-скрипт, который связан с другими модулями способом подключения модулей с методом открытого доступа. Проектирование каждого модуля происходит поэтапно, каждый модуль представляет собой логически завершенную часть программы.**

**Разработка приложения включает в себя следующие фазы:**

**Определение требований**

**Проектирование**

**Конструирование (также реализация либо кодирование)**

**Интеграция**

**Тестирование и отладка (также верификация)**

**Инсталляция**

**Поддержка**

**Для разработки приложения была выбрана модель проектирования, сочетающая в себе преимущества спиральной и каскадной моделей проектирования.**

**При каскадной модели проектирования процесс разработки выглядит как поток, последовательно проходящий фазы анализа требований, проектирования, реализации, тестирования, интеграции и поддержки.**

**Следуя каскадной модели, разработчик переходит от одной стадии к другой строго последовательно. Сначала полностью завершается этап «определение требований», в результате чего получается список требований к ПО. После того как требования полностью определены, происходит переход к проектированию, в ходе которого создаются документы, подробно описывающие для программистов способ и план реализации указанных требований. После того как проектирование полностью выполнено, программистами выполняется реализация полученного проекта. На следующей стадии процесса происходит интеграция отдельных компонентов, разрабатываемых различными командами программистов. После того как реализация и интеграция завершены, производится тестирование и отладка продукта; на этой стадии устраняются все недочёты, появившиеся на предыдущих стадиях разработки. После этого программный продукт внедряется и обеспечивается его поддержка — внесение новой функциональности и устранение ошибок. Тем самым, каскадная модель подразумевает, что переход от одной фазы разработки к другой происходит только после полного и успешного завершения предыдущей фазы, и что переходов назад либо вперёд или перекрытия фаз — не происходит.**

**Спиральная же модель подразумевает под собой постоянную доработку приложения и выпуск на каждом витке спирали новой версии программного продукта.**

### **3.2 ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ К ПРИЛОЖЕНИЮ**

**Программа должна быть применима для хранения и обработки данных призывника военкомата на военную службу. Она должна предоставлять удобный интерфейс для ввода и обработки данных. Предоставлять возможность изменения и сохранения полученных данных, выводить общие сведения и иметь возможность добавления и удаления сведений.**

**Программа должна обладать следующими свойствами:**

- масштабируемость;**
- иметь возможность редакции полученных расчетных данных, их сохранения;**
- безопасность и эффективность.**
- возможность дополнения**

**При определении этих целей важным стало стремление создать приложение, которое могло бы предоставлять возможность хранения и обработки данных призывника. Программа автоматизирует учета данных призывника военкомата.**

### **3.3 ЭТАПЫ РАЗРАБОТКИ ПРИЛОЖЕНИЯ**

**Разработка программного приложения для учета призывников военкомата включает в себя: создание удобного интерфейса пользователя, логичного и интуитивно понятного, чтобы**

**пользователь мог вводить исходные данные программы, такие как:**

- Исходные сведения о призывнике, соответствующие документу-перечню**
- Добавлять и изменять данные о призывнике**
- Удалять и дополнять данные призывника**
- Выводить общие данные о призывнике.**

**Программа сохраняет и обрабатывает данные о призывнике, и хранит это в базе данных MySQL. Пользователь программы может открыть страницу добавления нового призывника и ввести все необходимые данные, затем все сохранить.**

**Создание программы включает в себя несколько этапов:**

**На *первом этапе* происходит сбор и анализ данных о предметной области для которой разрабатывается программа.**

**На *втором этапе* разрабатывается интерфейс приложения.**

**На *третьем этапе* продумывается логика и алгоритм программы и взаимодействие компонентов системы.**

**На *четвертом этапе* происходит разработка модулей программы учета данных призывников для военкомата на языке PHP.**

**На *пятом этапе* происходит создание связей между модулями программы для военкомата.**

**На *шестом этапе* происходит разработка модуля вывода всех полученных данных**

**На *восьмом этапе* осуществляется тестирование и отладка приложения.**

### 3.4 ВЫБОР МОДЕЛИ СОЗДАНИЯ САЙТА

Программа обработки данных призывника спроектирована на блоках, все отдельные компоненты программы, разделены по отдельным блокам <div>

```
<body>
```

```
<div id="wrap">
```

```
<div id="title"></div>
```

```
<div id="menu">
```

```
<ul id="lmenu">
```

```
<p>Редактирование</p>
```

```
<li><a href="add_all.php">Добавить &raquo;</a> </li>
```

```
<li><a href="change_data_list.php">Изменить &raquo;</a> </li>
```

```
<li><a href="delete_data.php">Удалить &raquo;</a></li>
```

```
</ul>
```

```
<ul id="rmenu">
```

```
<p class="left">Призывники</p>
```

```
<li> <a href='../show_all_data.php?id=31'>&raquo; Тураходжаев  
Ильяс                   Нажмиддинович</li></a><li>                   <a
```

```
href='../show_all_data.php?id=37'>&raquo; Холматов Ислон  
Улугбекович</li></a><li>                   <a
```

```
href='../show_all_data.php?id=39'>&raquo; Хомидов Бахтиер  
Низамиддинович</li></a></ul>
```

```
<div id="rmenu" style="float:none; clear:both;">
```

```
<p>Поиск</p>
```

```
<h4>Введите данные для поиска</h4>
```

```
<form action="search.php" method="post" name="sform">
```

```

<center>
<select name="selects">
<option value="1">По фамилии</option>
<option value="2">По имени</option>
<option value="3">По учетному номеру</option>
</select>
</center><p align="center">
<input name="sqquery" style="position:relative; left:0;"
type="text" size="15" maxlength="30" />
<input name="submit" style="position:relative; left:0;"
type="submit" value="Искать" />
</form>
</p>
</div></div>

```

1. Первый блок – это верхний заголовок программы, он считается описательным, и говорит о содержании и применении программы.

### Рис 1. Верхний блок программы

Этот блок содержит описание главной страницы и следующий код.

```
<div id="title"></div> описание вставки блока в программу
```

Для оформления этого блока применяется таблица CSS

```
#title{  
width:100%; height:150px; - ширина и высота блока заголовка  
background:#C0FC25; - задний цветовой фон  
background-image:url(BG.png); - картинка на заднем плане  
border-top:1px solid #92A813; - оформление рамки заголовка  
border-radius:20px; - радиус округления рамки  
}
```

### 3.5 СОЗДАНИЕ МЕНЮ ПРОГРАММЫ



Рис 2. Внешний вид меню программы

В меню можно добавлять, редактировать и удалять сведения о призывниках, каждый пункт кнопки меню обрабатывается скриптом php.

Кнопка **Добавить** содержит обработчик события нажатия на кнопку

```
<p>Редактирование</p>
<li><a href="add_all.php">Добавить &raquo;</a> </li>
<li><a href="change_data_list.php">Изменить &raquo;</a> </li>
<li><a href="delete_data.php">Удалить &raquo;</a></li>
</ul>
```

Меню сайта создано с помощью тэга `ul – li`

Обработчик события нажатия на кнопку указан в описании, скрипт, обрабатывающий нажатие кнопки **Добавить** содержится в файле `add_all.php`

Код скрипта:

```
<body>
```

```
<div id="wrapn">
<h1>Учетная карточка призывника</h1>
<h2>Ниже введите личные данные призывника в поля (*
обязательные поля)</h2>

<form      action="add/add_all_data.php"      method="post"
name="form1">
<p>Учетный номер призывника:</p>
<input      name="Uch_nomer"      type="text"      size="30"
maxlength="30" />
<br/> <br/>
<p>Фамилия: </p>
<input name="Fam" type="text" size="30" maxlength="30" />
<br/> <br/>
<p>Имя: </p>
<input name="Имя" type="text" size="30" maxlength="30" />
<br/> <br/>
<p>Отчество: </p>
<input name="Otch" type="text" size="30" maxlength="30" />
<br/> <br/>
<p>Серия паспорта: </p>
<input name="Seria_pass" type="text" size="10" maxlength="10"
/>
<br/> <br/>
<p>Номер паспорта: </p>
<input name="N_pass" type="text" size="20" maxlength="20" />
```

<br/> <br/>

<p>Дата выдачи: </p>

<input name="Data\_vid\_pass" type="text" size="20" maxlength="20" />

<br/> <br/>

<p>Кем выдан: </p>

<input name="Kem\_vid\_pass" type="text" size="30" maxlength="30" />

<br/> <br/>

<p>Фото: </p>

<input name="Foto" type="file" />

<br/> <br/>

<h1>Общие сведения о призывнике</h1>

<p>Дата рождения: </p>

<input name="data\_rojd" type="text" size="30" maxlength="30" />

<br/> <br/>

<p>Место рождения: </p>

<input name="mesto\_rojd" type="text" size="30" maxlength="30" />

<br/> <br/>

<p>Национальность: </p>

<input name="Nacionalnost" type="text" size="30" maxlength="30" />

<br/> <br/>

<p>Адрес места жительства: </p>

```
<input name="Adres_mesta_jit" type="text" size="30"
maxlength="30" />
```

```
<br/> <br/>
```

```
<p>Социальное происхождения: </p>
```

```
<input name="Soc_proishoj" type="text" size="30"
maxlength="50" />
```

```
<br/> <br/>
```

```
<p>Родной язык: </p>
```

```
<input name="Rodnoi_yazik" type="text" size="20"
maxlength="20" />
```

```
<br/> <br/>
```

```
<p>Владение государственным <br /> языком: </p>
```

```
<input name="Vlad_gos_yaz" type="text" size="20"
maxlength="20" />
```

```
<br/> <br/>
```

```
<p>Владение иностранным языком: </p>
```

```
<input name="Vlad_inostr_yaz" type="text" size="30"
maxlength="30" />
```

```
<br/> <br/>
```

```
<p>Место учебы:</p>
```

```
<input name="Mesto_uch" type="text" size="30" maxlength="50"
/>
```

```
<br/> <br/>
```

```
<p>Срок обучения: </p>
```

```
<input name="Srok_obuch" type="text" size="10"
maxlength="20" />
```

<br/> <br/>

<p>Номер курса: </p>

<input name="Nomer\_kursa" type="text" size="10" maxlength="20" />

<br/> <br/>

<p>Дата окончания учебы: </p>

<input name="Data\_okonch\_uch" type="text" size="30" maxlength="30" />

<br/> <br/>

<p>Специальность:</p>

<input name="Specialnost" type="text" size="30" maxlength="30" />

<br/> <br/>

<p>Образование: </p>

<input name="Obrazovanie" type="text" size="30" maxlength="30" />

<br/> <br/>

<p>Место работы: </p>

<input name="Mesto\_rab" type="text" size="30" maxlength="30" />

<br/> <br/>

<p>Гражданская профессия: </p>

<input name="Grajd\_prof" type="text" size="30" maxlength="30" />

<br/> <br/>

<p>Вид спорта: </p>

```
<input name="Vid_sporta" type="text" size="30" maxlength="10" />
```

```
<br/> <br/>
```

```
<p>Разряд по спорту: </p>
```

```
<input name="Razr_po_sportu" type="text" size="10" maxlength="20" />
```

```
<br/> <br/>
```

```
<p>С каким государством <br /> имеет связь: </p>
```

```
<input name="S_kakim_gos_im_svyaz" type="text" size="30" maxlength="20" />
```

```
<br/> <br/>
```

```
<p>С каким гос-вом имеют <br />связь родственники: </p>
```

```
<input name="S_kakim_gos_im_svyaz_rodstv" type="text" size="30" maxlength="30" />
```

```
<br/> <br/>
```

```
<p>Судимость:</p>
```

```
<input name="Sudimost" type="text" size="30" maxlength="30" />
```

```
<br/> <br/>
```

```
<p>Судимость родственников: </p>
```

```
<input name="Sudimost_rodstv" type="text" size="30" maxlength="20" />
```

```
<br/> <br/>
```

**В коде указано какие поля необходимо заполнять, при нажатии на кнопку добавить, появляется следующее окно:**

**ОТДЕЛ ПО ДЕЛАМ ОБОРОНЫ  
ГОРОДА ФЕРГАНЫ**

**Учетная карточка призывника**

Ниже введите личные данные призывника в поля (\* обязательные поля)

Учетный номер призыва:

Фамилия:

Имя:

Отчество:

Серия паспорта:

Номер паспорта:

**Редактирование**

Главная »

Добавить »

Изменить »

Удалить »

**Призывники**

» [Тухраходжаев Ильяс  
Нажмиддинович](#)

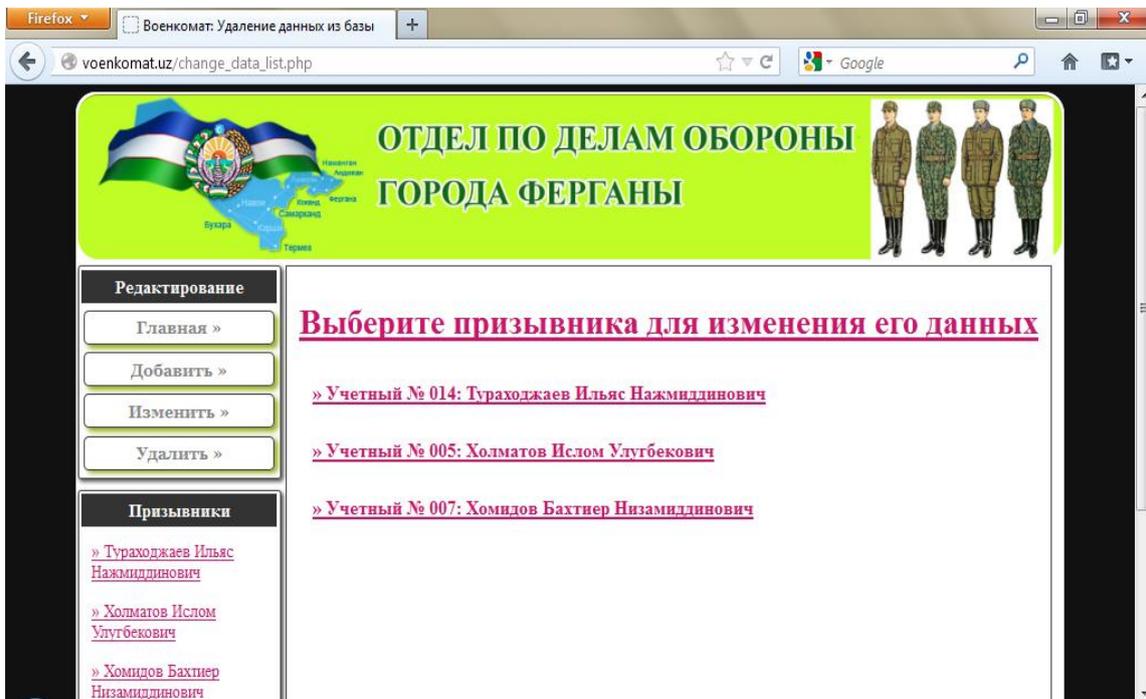
» [Холматов Ислам  
Улугбекович](#)

» [Хомидов Бахтиер  
Низамиддинович](#)

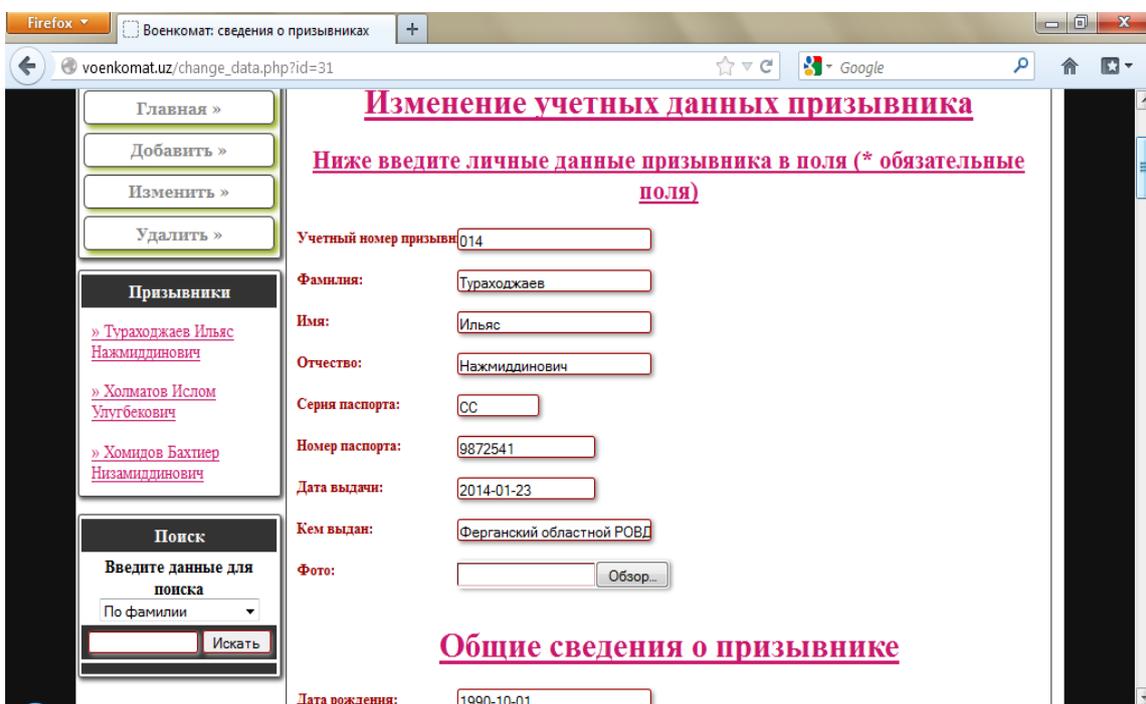
**Рис 4. В окне выводится форма, со всеми необходимыми полями, их надо заполнить данными.**

**После заполнения полей необходимо нажать на кнопку “Сохранить”.**

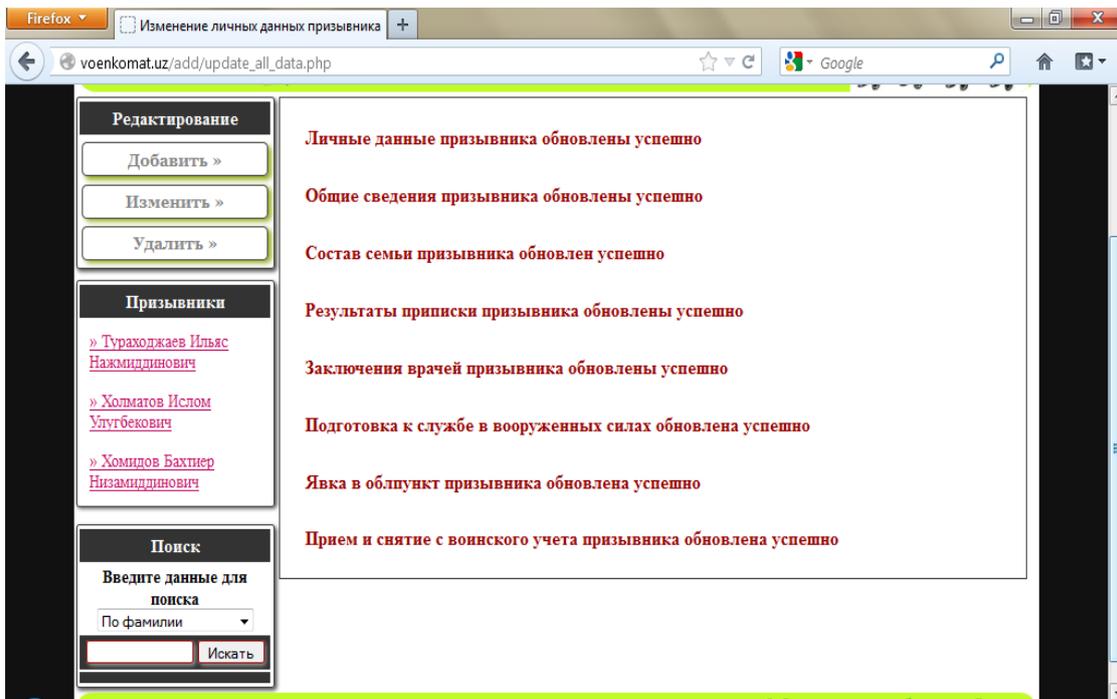
**Кнопка Изменить необходима для изменения данных о призывнике, в форму подгружаются данные из базы данных MySQL для конкретного призывника, при нажатии на кнопку появляется следующее окно:**



**Рис 4. Окно программы изменения данных о призывнике**  
**Для того чтобы изменить данные о призывнике, надо выбрать его имя и откроется окно редактирования данных:**



**Рис 5. Окно редактирования введенных данных призывника**  
**При изменении данных, необходимо нажать кнопку Сохранить изменения, и обновленные данные попадут в базу:**



**Рис 7. Результаты изменения данных**

**Скрипт, производящий редактирование данных содержится в файле `change_data_list.php`**

**Код обработчика кнопки:**

```

<body>
<div id="wrap">
<div id="title"></div>
<div id="menu">
<ul id="lmenu">
<p >Редактирование</p>
<li><a href="index.php"> Главная &raquo;</a> </li>
<li><a href="add_all.php">Добавить &raquo;</a> </li>
<li><a href="change_data_list.php">Изменить &raquo;</a> </li>
<li><a href="delete_data.php">Удалить &raquo;</a></li>
</ul>
<? include("blocks/pmenu.php") ?>

```

```

</div>
<style type="text/css">
<div id="wrapn">
<h1 >Выберите призывника для изменения его данных</h1>
<?
@mysql_query("SET names 'UTF8'"); // обращение к базе
данных за сведениями
$result = mysql_query("SELECT Uch_nomer,Fam, Imya, Otch, id
FROM Lich_dannie ORDER BY id ASC",$db);
$myrow = mysql_fetch_array($result); // выбор полей из базы для
загрузки и сохранения
do
{
printf ("<p><a href='change_data.php?id=%d'>&raquo; Учетный
№           %s:           %s           %s
%s</a></p>", $myrow["id"], $myrow["Uch_nomer"], $myrow["Fa
m"], $myrow["Imya"], $myrow["Otch"]);
}
while ($myrow = mysql_fetch_array($result));
?>
</div>
<div id="footer">&reg; Отдел по делам обороны г.
Ферганы</div>
</div>
</body>
</html>

```

При нажатии на кнопку изменить происходит изменение данных о призывнике в базе и сохранение изменений

При нажатии на кнопку Удалить появляется следующее окно:

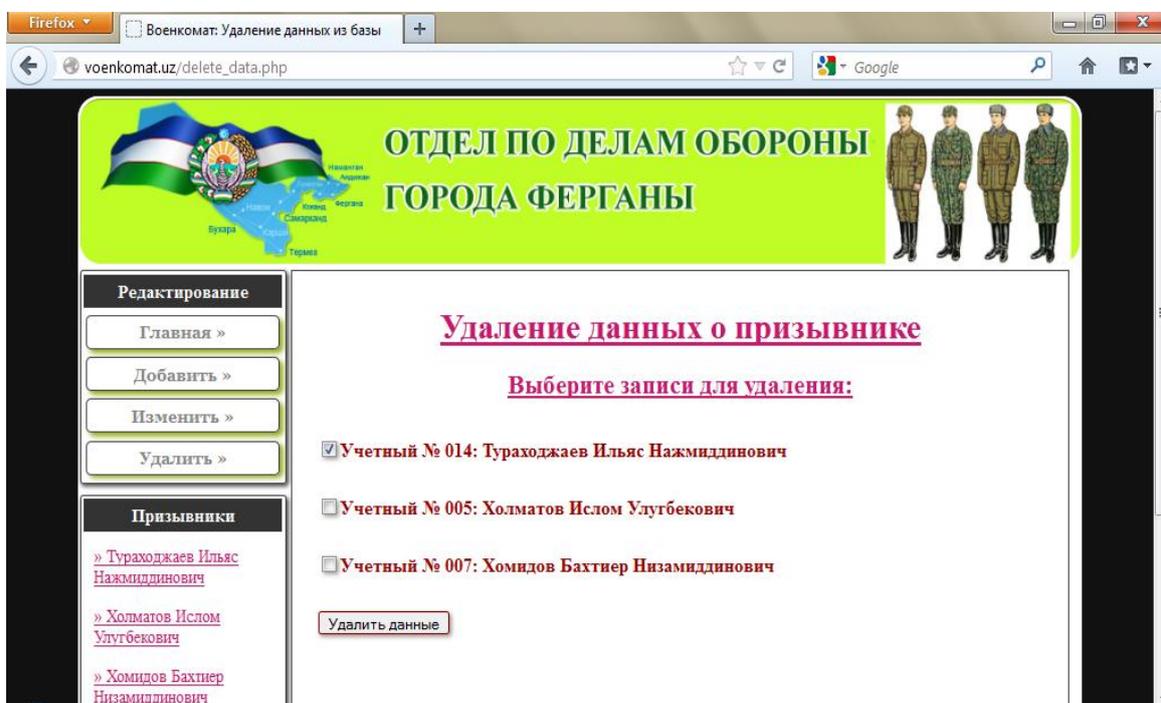


Рис 8. Окно удаления данных о призывнике

При удалении данных надо выбрать галочкой, какого призывника необходимо удалить. Удаление данных осуществляется скриптом:

```
<h1>Удаление данных о призывнике</h1>
```

```
<h2>Выберите записи для удаления:</h2>
```

```
<form action="add/delete_selected_data.php" method="post">
```

```
<?php
```

```
    $result = mysql_query("SELECT  
id,Uch_nomer,Fam,Imya,Otch FROM Lich_dannie ORDER BY  
id");
```

```

        $myrow = mysql_fetch_array($result);
    do
    {
        echo '<p><input type="checkbox" name="item[]"
value="".$myrow['id'].'" />';
        echo "Учетный № ".$myrow['Uch_nomer'].": "; // выбор
записей для удаления
        echo $myrow['Fam']." ";
        echo $myrow['Imya']." ";
        echo $myrow['Otch'];
        echo '</p>';
    }
    while ($myrow = mysql_fetch_array($result)); // вывод
данных удаленных призывников
    ?>
    <p><input name="submit" type="submit" value="Удалить
данные" /></p>
</form>
</div>
<div id="footer">&reg; Отдел по делам обороны г.
Ферганы</div>
</div>
</body>

```

### 3.6 СОЗДАНИЕ БЛОКА ВЫВОДА ЗАПИСАННЫХ ПРИЗЫВНИКОВ

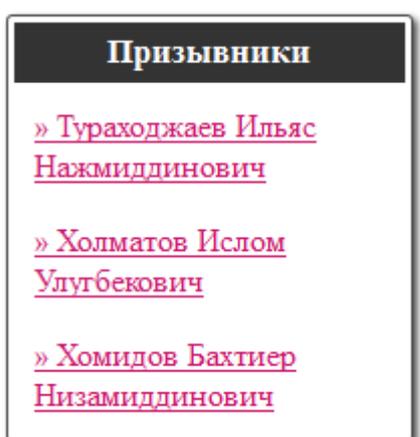


Рис 9. Блок вывода сведений о призывниках.

При нажатии на пункты вывода сведений о призывниках, выводится подробная информация о призывнике в форму и открывается следующее окно:

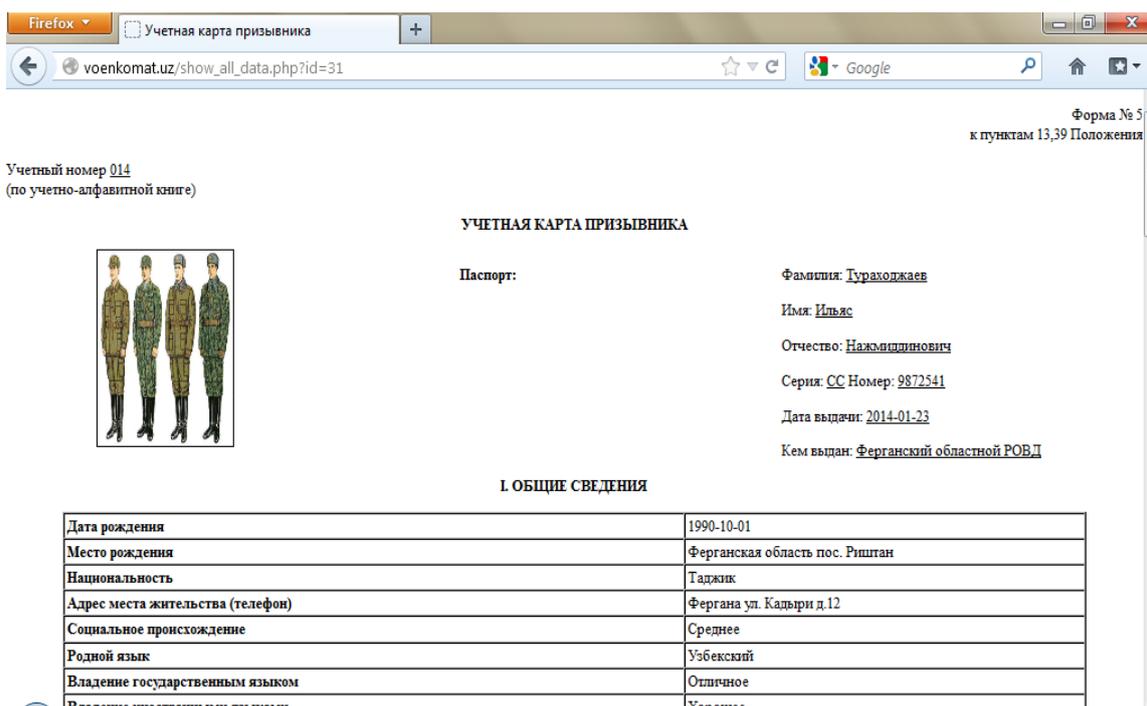


Рис 10. Данные призывника из таблицы

Соответствующий блок имеет такой код

```
<ul id="pmenu">
<p class="left">Призывники</p>
<li> <a href='../show_all_data.php?id=31'>&raquo;
    Тураходжаев        Ильяс        Нажмиддинович</li></a><li> <a
href='../show_all_data.php?id=37'>&raquo;
    Холматов          Ислом        Улугбекович</li></a><li> <a
href='../show_all_data.php?id=39'>&raquo;
```

Хомидов Бахтиер Низамиддинович</li></a>

</ul>

Он создан как список призывников

При выборе пункта из этого блока обработка осуществляется скриптом:

```
<body style="background:#FFF;">
```

```
<?
```

```
$result1 = mysql_query ("SELECT * FROM Lich_dannie WHERE id = $id", $db);
```

```
$myrow = mysql_fetch_array($result1);
```

```
$Uch_nomer = $myrow['Uch_nomer'];
```

```
$Fam = $myrow['Fam'];
```

```
$Imya = $myrow['Imya'];
```

```
$Otch = $myrow['Otch'];
```

```
$Seria_pass = $myrow['Seria_pass'];
```

```
$N_pass = $myrow['N_pass'];
```

```
$Data_vid_pass = $myrow['Data_vid_pass'];
```

```
$Kem_vid_pass = $myrow['Kem_vid_pass'];
```

```
$Foto = $myrow['Foto'];
```

```
?>
```

```
<div id="swrap">
```

```
<p align="right" >Форма № 5 <br>к пунктам 13,39 Положения</p>
```

```
<p align="left" >Учетный номер <u><? echo $Uch_nomer?></u> <br>(по учетно-  
алфавитной книге)</p>
```

```
<p class="header">Учетная карта призывника</p>
```

```

```

```
<p id="pass">Паспорт:</p>
```

```
<div id="pass_d">
```

```
<p>Фамилия: <u><? echo $Fam?></u></p>
```

```
<p>Имя: <u><? echo $Imya?></u></p>
```

```
<p>Отчество: <u><? echo $Otch?></u></p>
```

```
<p>Серия: <u><? echo $Seria_pass?></u> Номер: <u><? echo $N_pass?></u></p>
```

```
<p>Дата выдачи: <u><? echo $Data_vid_pass?></u></p>
```

```
<p>Кем выдан: <u><? echo $Kem_vid_pass?></u></p>
```

```
</div>
```

```
<? // Выборка из базы данных сведений о призывнике для соответствующих полей  
базы.
```

```
$result2 = mysql_query ("SELECT * FROM Obshie_dannie WHERE id_prizivnika =  
$id", $db);
```

```
$myrow1 = mysql_fetch_array($result2);
```

```
$data_rojd = $myrow1['data_rojd'];
```

```
$mesto_rojd = $myrow1['mesto_rojd'];
```

```
$Nacionalnost = $myrow1['Nacionalnost'];
```

```
$Adres_mesta_jit = $myrow1['Adres_mesta_jit'];
```

```
$Soc_proishoj = $myrow1['Soc_proishoj'];
```

```
$Rodnoi_yazik = $myrow1['Rodnoi_yazik'];
```

```
$Vlad_gos_yaz = $myrow1['Vlad_gos_yaz'];
```

```
$Vlad_inostr_yaz = $myrow1['Vlad_inostr_yaz'];
```

```
$Mesto_uch = $myrow1['Mesto_uch'];
```

```
$Srok_obuch = $myrow1['Srok_obuch'];
```

```
$Nomer_kursa = $myrow1['Nomer_kursa'];
```

```
$Data_okonch_uch = $myrow1['Data_okonch_uch'];
```

```
$Specialnost = $myrow1['Specialnost'];
```

```
$Obrazovanie = $myrow1['Obrazovanie'];
```

```
$Mesto_rab = $myrow1['Mesto_rab'];
```

```
$Grajd_prof = $myrow1['Grajd_prof'];
```

```
$Vid_sporta = $myrow1['Vid_sporta'];
```

```
$Razr_po_sportu = $myrow1['Razr_po_sportu'];
```

```
$S_kakim_gos_im_svyaz = $myrow1['S_kakim_gos_im_svyaz'];
```

```
$S_kakim_gos_im_svyaz_rodstv = $myrow1['S_kakim_gos_im_svyaz_rodstv'];
```

```
$Sudimost = $myrow1['Sudimost'];
```

```
$Sudimost_rodstv = $myrow1['Sudimost_rodstv'];
```

```
?>
```

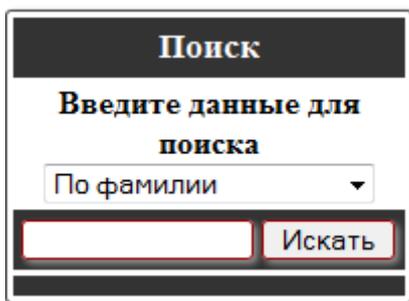


Рис 11. Окно поиска данных о призывнике

Для того, чтобы осуществить поиск при большом объеме данных из базы в программе предусмотрена форма поиска данных. Форма поиска имеет следующий код:

```
<form action="../search.php" method="post" name="sform">
<center> Данные с формы передаются файлу скрипта search.php методом post
<select name="selects">
<option value="1">По фамилии</option>
<option value="2">По имени</option>
<option value="3">По учетному номеру</option>
</select>
</center><p align="center">
<input name="squery" style="position:relative; left:0;" type="text" size="15"
maxlength="30" />
<input name="submit" style="position:relative; left:0;" type="submit"
value="Искать" />
</form>
```

При вводе строки поиска в форму и выборе поля поиска, необходимо нажать кнопку Поиск

И запустится соответствующий обработчик события, содержащийся в файле search.php

```
<?
include("blocks/bd.php");
@mysql_query("SET NAMES 'UTF8'");
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link rel="stylesheet" type="text/css" href="../images/styles.css" />
<title>Результаты поиска</title>
</head>
<body>
<div id="wrap">
<div id="title"></div>
<div id="menu">
<ul id="lmenu">
<p>Редактирование</p>
<li><a href="add_all.php">Добавить &raquo;</a> </li>
<li><a href="change_data_list.php">Изменить &raquo;</a> </li>
<li><a href="delete_data.php">Удалить &raquo;</a></li>
</ul>
<? include("blocks/pmenu.php") ?>
</div>
<p id="text">
<style type="text/css">
h1,h2 {color:#CA176C; text-decoration:underline; text-align:center;}
p {padding:0; margin:0 10px 0 0 ; color:#900; font:bold 14px "Times New
Roman", Times, serif; }
#wrapn{ // Оформление страницы сайта через CSS
min-height:350px;
margin:5px;
float:left;
```

```

width: 740px;
border:1px solid #424242;
padding:10px;}
#wrapn p{padding:15px; font-size:18px;}
#wrapn p a{color:#CA176C;}
#wrapn p a:hover{text-decoration:none;}
input {border:1px solid #900; border-radius:3px; box-shadow:2px 2px 3px
#999999; }
#submit{position:relative; left:0; float:right;}
</style>
<? include ("lock.php")?> // подключение замка программы и авторизации
<div id="wrapn">
<h1>Результаты поиска</h1>
<?
$query = $_POST["squery"];
$query = trim($query);
$sel = $_POST["selects"];
include("blocks/bd.php");
@mysql_query("SET names 'UTF8'");
if($sel == "1")
{
$result1 = mysql_query ("SELECT id, Uch_nomer, Fam, Imya, Otch FROM
Lich_dannie WHERE Fam LIKE '%$query%'", $db);
$myrow = mysql_fetch_array($result1);
}
if($sel == "2")
{
$result1 = mysql_query ("SELECT id, Uch_nomer, Fam, Imya, Otch FROM
Lich_dannie WHERE Imya LIKE '%$query%'", $db);
$myrow = mysql_fetch_array($result1);
}
if($sel == "3")
{
$result1 = mysql_query ("SELECT id, Uch_nomer, Fam, Imya, Otch FROM
Lich_dannie WHERE Uch_nomer LIKE '%$query%'", $db);
$myrow = mysql_fetch_array($result1);
}
$i=1;
do
{
if($query == ""){printf("<p>Вы ничего не ввели в поле запроса</p>"); break;}
if(mysql_num_rows($result1)==0) {printf("<p>По запросу ничего не
найден</p>"); break;}
printf ("<p><a href='show_all_data.php?id=%d'>%d) %s %s
%s</a></p><br>", $myrow["id"], $i, $myrow["Fam"], $myrow["Imya"], $myrow["Otch"]);
$i++;
}
while ($myrow = mysql_fetch_array($result1));
?>
</div>
</p>
<div id="footer">&reg; Отдел по делам обороны г. Ферганы</div>
</div>
</body>
</html>

```

### 3.7 ОПИСАНИЕ АЛГОРИТМА ПРОГРАММЫ

**При запуске программы запрашиваются необходимые данные о призывнике, для добавления их в базу MySQL. После**

нажатия на кнопку **Добавить** выводится окно с полями ввода данных. Есть возможность изменения и удаления сведений, общий вывод сведений. Весь алгоритм работы программы можно представить в следующем виде:

**Запуск.**

**Ввод запрашиваемых данных.**

**Сохранение данных в базу MySQL.**

**Вывод сведений о призывнике.**

**Редактирование сведений.**

**Удаление сведений о призывнике.**

**Код главной страницы программы:**

```
<?
include("blocks/bd.php");
@mysql_query("SET NAMES 'UTF8'");
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<link rel="stylesheet" type="text/css" href="../images/styles.css" />
<title>Военкомат: сведения о призывниках</title>
</head>
```

```

<body>
<div id="wrap">
<div id="title"></div>
<div id="menu">
<ul id="lmenu">
<p>Редактирование</p>
<li><a href="add_all.php">Добавить &raquo;</a> </li>
<li><a href="change_data_list.php">Изменить &raquo;</a> </li>
<li><a href="delete_data.php">Удалить &raquo;</a></li>
</ul>
<? include("blocks/pmenu.php") ?>
</div>
<p id="text">
<style type="text/css">
h1,h2 {color:#CA176C; text-decoration:underline; text-align:center;}
p {padding:0; margin:0 10px 0 0 ; color:#900; font:bold 14px "Times New Roman", Times, serif; }
#wrapn{
min-height:350px;
margin:5px;
float:left;
width: 740px;
border:1px solid #424242;
padding:10px;}

```

```

#wrapn p{padding:15px; font-size:18px;}
#wrapn p a{color:#CA176C;}
#wrapn p a:hover{text-decoration:none;}
input {border:1px solid #900; border-radius:3px; box-shadow:2px
2px 3px #999999; }
#submit{position:relative; left:0; float:right;}
</style>
<? //include ("lock.php")?>
<div id="wrapn">
<h1 >Главная страница отдела по дела обороны г.
Ферганы</h1>
<h2>Выберите призывника для просмотра его данных</h2>
<?
@mysql_query("SET names 'UTF8'");
$result = mysql_query("SELECT Uch_nomer,Fam, Imya, Otch, id
FROM Lich_dannie ORDER BY id ASC",$db);
$myrow = mysql_fetch_array($result);
do
{
printf ("<p><a href='show_all_data.php?id=%d'>&raquo;
Учетный № %s: %s %s
%s</a></p>",$myrow["id"],$myrow["Uch_nomer"],$myrow["Fa
m"],$myrow["Imya"],$myrow["Otch"]);
}
while ($myrow = mysql_fetch_array($result));
?>

```

**</div>**

**</p>**

**<div id="footer">&reg; Отдел по делам обороны г.  
Ферганы</div>**

**</div>**

**</body>**

**</html>**

### 3.8 ОФОРМЛЕНИЕ СТРАНИЦЫ

**@charset "utf-8";**

**body {background:#111; padding:0; margin:0;}**

**#wrap{ // Оформление шаблона программы**

**padding:1px;**

**width:980px;**

**background:#fff;**

**margin:7px auto;**

**border:1px solid #fff;**

**border-radius:20px;**

**}**

**#title{ // Оформление заголовка программы**

**width:100%; height:150px;**

**background:#C0FC25;**

**background-image:url(BG.png);**

**border-top:1px solid #92A813;**

**border-radius:20px;**

**}**

```

#menu{float:left; }
#lmenu{ // Оформление меню программы
border-radius: 3px;
box-shadow:2px 2px 3px #444;
width:200px;
border:1px solid #424242;
padding:0px; margin:5px 0;
}
#lmenu li{
border:1px solid #444;
border-radius:5px;
box-shadow:3px 3px 4px #92A813;
padding:5px 8px; list-style:none; margin:7px 5px; text-
align:center;
}
#lmenu li a{text-decoration:none; display:block;
color:#888;
font-weight:bold;
font-family:Georgia, "Times New Roman", Times, serif}
#lmenu li a:hover{color:#fff; background:#92A813;}
#lmenu p{ // Оформление вывода сведений о призывниках
margin:3px; padding: 5px 10px; background:#333; font-
size:17px; color:#fff; text-align:center; }
#pmenu{
width:200px;
border-radius: 3px;

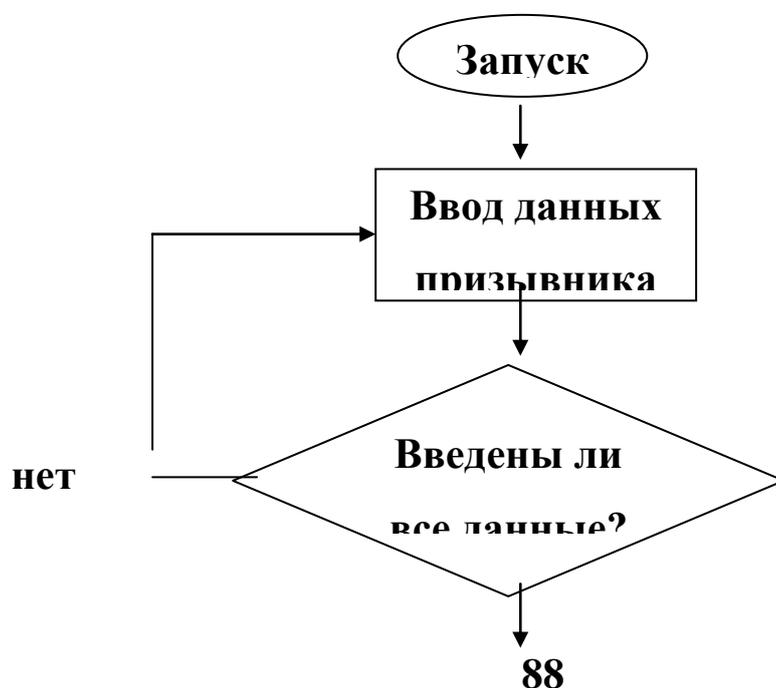
```

```

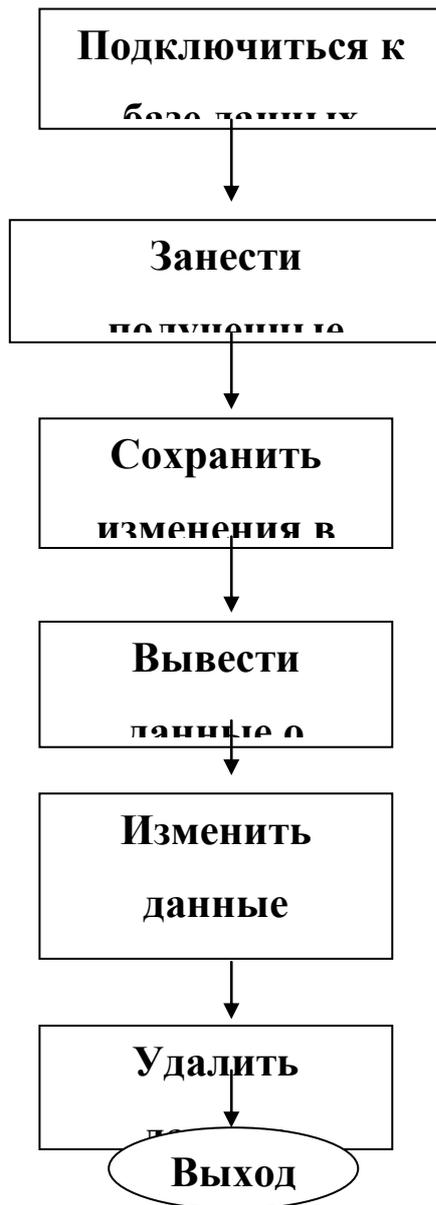
box-shadow:2px 2px 3px #444;
border:1px solid #424242;
margin-top:5px;
padding:0;
float:left;}
#pmenu li{
    padding:5px 8px; list-style:none; margin:7px 5px;
    color:#D31616;
}
#pmenu li a{color:#CA176C; text-decoration:underline;}
#pmenu li a:hover{text-decoration:none;}
#pmenu p{
    float:none;
    font-size:17px;
    margin:3px; padding: 5px 0px; background:#333; color:#fff;
text-align:center;
}
#pmenu h4{margin:0; padding:0; text-align:center}
#text{float:right;}
.left{float:left;}
#footer{position:relative; top:5px; clear:both; width:99.9%;
height:30px; color:#36810A; border:1px solid ##C0FC25; border-
radius:20px; background:#C0FC25; font-weight:bold; text-
align:right;}
// Оформление нижнего графического блока

```

```
#swrap{float:right; width:100%; border:1px solid #333;
font:normal 18px "Times New Roman", Times, serif;
background:#0FC; padding:5px;}
.header{text-align:center; font-weight:bold; text-
transform:uppercase; width:100%;}
.left{float:left;}
.right{float:right;}
.clear{clear:both;}
#pass{text-indent:350px; font-weight:bold; float:left;}
#pass_d{width:400px; float:right;}
#formaN{ float:none; width:200px; text-align:center;}
#uchN{ width:200px; }
.table{text-align:left;}
.padmarg0{padding:0; margin:0;}
img{position:relative; left:100px;}
```



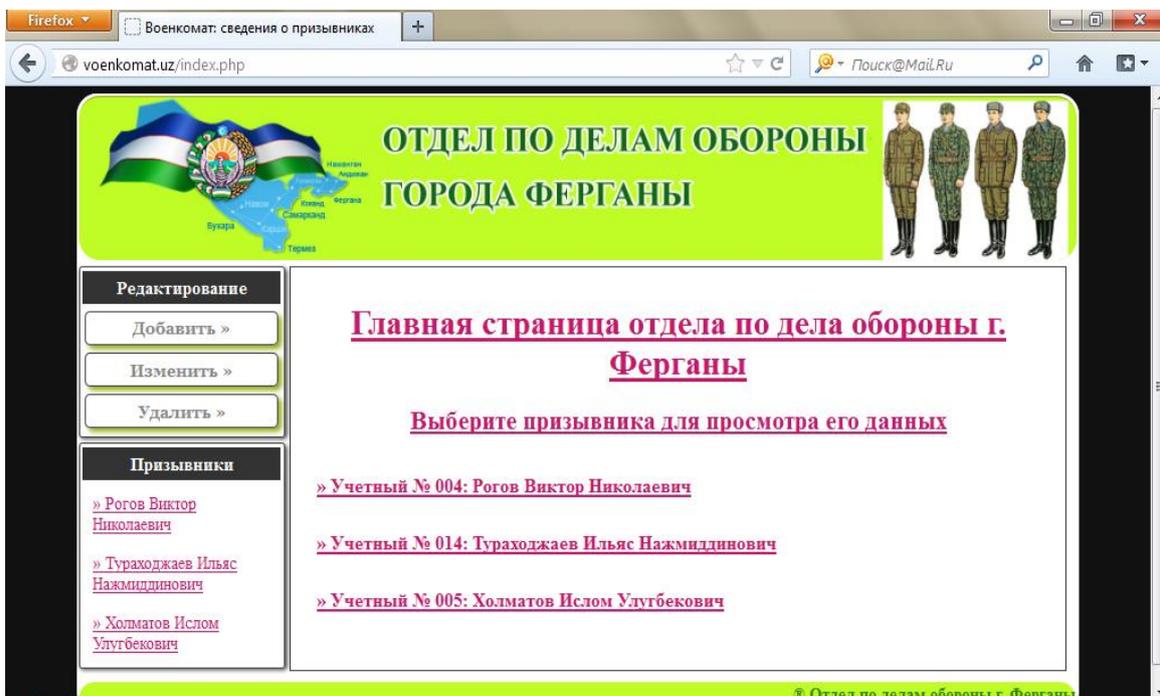
да



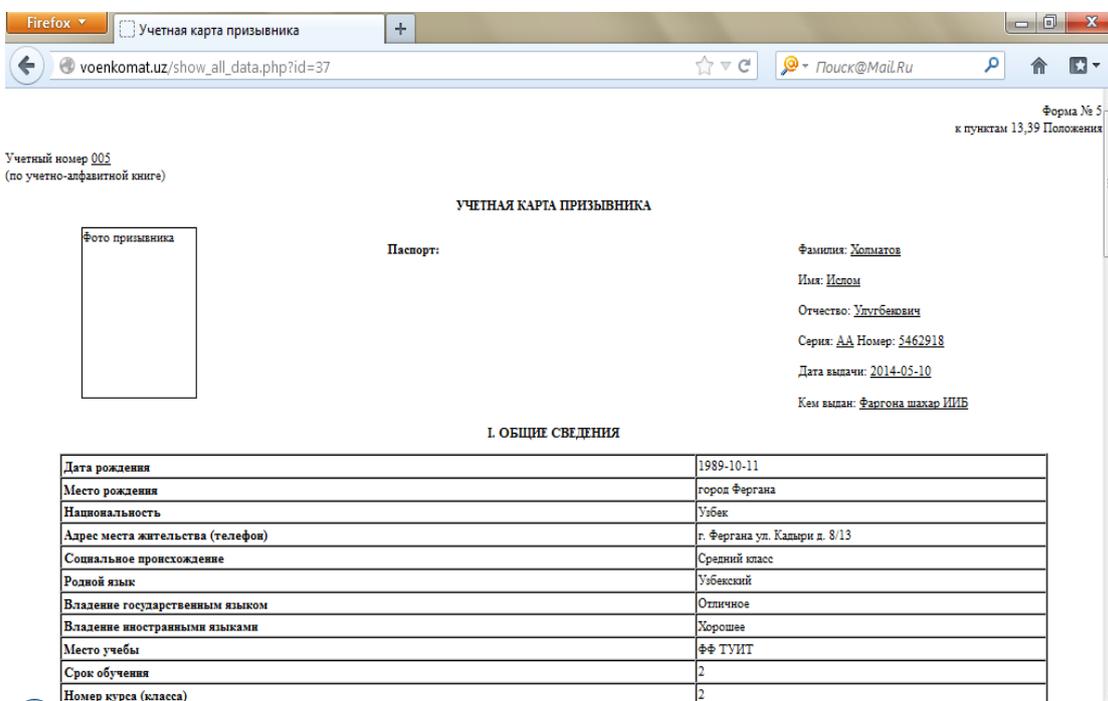
**Рисунок 3.1. Укрупнённая блок-схема программы**

### 3.9 ОПИСАНИЕ РАБОТЫ ПРОГРАММЫ

Для запуска программы необходимо перейти в браузер и набрать в адресной строке [voenkomat.uz](http://voenkomat.uz) и откроется следующее окно браузера:



**Рис 12. Главное окно программы сведения о призывниках**  
 Здесь в левой части экрана находится меню управления данными и список призывников, в центральной части можно выбрать и перейти на страницу личных данных призванного:



**Рис 13. Вывод сведений о призваннике в личную карту**  
 В этом окне можно просмотреть все данные призванного из личной карты

Для добавления сведений о новом призывнике необходимо выбрать из меню пункт **Добавить** и появится следующее окно:

Firefox | Военкомат: сведения о призывниках | voenkomat.uz/add\_all.php | Поиск@Mail.Ru

**ОТДЕЛ ПО ДЕЛАМ ОБОРОНЫ  
ГОРОДА ФЕРГАНЫ**

**Редактирование**

Главная »  
Добавить »  
Изменить »  
Удалить »

**Призывники**

» [Рогов Виктор Николаевич](#)  
» [Тураходжаев Ильяс Нажмиддинович](#)  
» [Холматов Ислон Улугбекович](#)

**Учетная карточка призывника**

Ниже введите личные данные призывника в поля (\* обязательные поля)

Учетный номер призывника:

Фамилия:

Имя:

Отчество:

Серия паспорта:

Номер паспорта:

Рис 14. Ввод данных о новом призывнике

Все данные необходимо ввести в поля и нажать кнопку **Сохранить**.

Если необходимо затем изменить данные, выберите из меню пункт **Изменить** и появится следующее окно:

Firefox | Военкомат: Удаление данных из базы | voenkomat.uz/change\_data\_list.php | Поиск@Mail.Ru

**ОТДЕЛ ПО ДЕЛАМ ОБОРОНЫ  
ГОРОДА ФЕРГАНЫ**

**Редактирование**

Главная »  
Добавить »  
Изменить »  
Удалить »

**Призывники**

» [Рогов Виктор Николаевич](#)  
» [Тураходжаев Ильяс Нажмиддинович](#)  
» [Холматов Ислон Улугбекович](#)

**Выбор призывника для изменения его данных**

» [Учетный № 004: Рогов Виктор Николаевич](#)  
» [Учетный № 014: Тураходжаев Ильяс Нажмиддинович](#)  
» [Учетный № 005: Холматов Ислон Улугбекович](#)

Рис 15. Выбор данных о призывнике, которые надо изменить

Выберите призывника, чьи данные вы хотите изменить и откроется следующее окно:

Рис 16. Поля ввода измененных данных о призывнике

После введения изменений также необходимо нажать кнопку Сохранить.

Для удаления данных о призывнике, выберите из меню пункт Удалить и выберите записи для удаления:

## **Рис 17. Окно удаления данных о призывнике**

**Затем нажмите Удалить и данные и данные из базы будут удалены.**

**Описанная программа, соответствует поставленным требованиям, и обладает всеми необходимыми качествами, кроме того имеет удобный и интуитивно понятный интерфейс. Не вызывает затруднений при использовании.**

### **ЗАКЛЮЧЕНИЕ**

**В ходе выполнения выпускной квалификационной работы был получен полнофункциональная база данных, полностью готовый к применению. С его помощью пользователи смогут получать необходимую информацию. При размещении его в глобальной сети география распространения возрастает до масштабов всего мира.**

**При разработке базы данных были проанализированы современные web-технологии, позволяющие создавать интерактивные web-страницы. Наиболее подходящими для выполнения поставленной задачи оказались Macromedia Dreamweaver MX 2004 и 1.0.12 PHP.**

**Разработанная база данных удовлетворяет всем требованиям, поставленным на этапе постановки задачи. При разработке базы данных были использованы готовые модули аутентификации, форума и осуществления поиска по сайту. Данные модули были доработаны с учетом**

**специфики web-базы данных и успешно внедрены в его структуру.**

**В качестве дальнейшего совершенствования базы данных представляется возможным разработка модулей доступа к системам дистанционного образования. Так же возможна доработка интерфейса базы данных с целью дальнейшего повышения его информативности, привлекательности и удобства.**

**Поставленные цели и задачи выполнены и разработанный модуль вполне можно применять. Программа установлена Ферганком горадском в настоящее время проходит обрабацию в Ферганком горадском ваенкомате имеются акт о внедрении программногo продукция.**

**Так как структура дистанционных учебных курсов подразумевает модульный принцип, то разработанный модуль может быть включен в состав курса по Интернет технологиям, или использоваться отдельно.**

## СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

### Использованные источники

1. **Web Database Application with PHP and MySQL, 2nd Edition** By David Lane, Hugh E. Williams. © O'Reilly, May 2004. ISBN:0-596-005431.
2. **CMS List. Обзор cms. Сайт о системах управления сайтом.** <http://www.cmslist.ru>
3. **CMS по-русски.** <http://portal.ru>
4. **Материал из Википедии — свободной энциклопедии о системах управления сайтом.** <http://ru.wikipedia.org/wiki/CMS>
5. **Content management system** <http://www.brutto.ru/informacija/uznat-bolshe/content-management-system>
6. **инструменты.** <http://docs.joom.ru/!/;C651=00:Categories>
7. **CMS обзор: CMS, движок базы данных, система управления сайтом, mambo, php nuke, netcat, phpbb, invision power board, vbulletin.** <http://cmsobzor.ru/news.php>
8. **Borland InterBase Workgroup Server. API Guide.** - Borland International Inc, 1995 - 330 с.
- 9 **Borland InterBase Workgroup Server. DataDefinition Guide.** - Borland International Inc, 1995 - 212 с.
- 11 **Borland InterBase Workgroup Server. Language Reference.** - Borland International Inc, 1995 - 234 с.

**12 Borland InterBase Workgroup Server. Programmer's Guide. - Borland International Inc, 1995 - 340 с.**

**Microsoft Online Documentation: Win32 Programmers Reference.**

**14 R.Barker "CASE\* Method - Entity Relationship Modelling". - Oracle Inc., 1990 - 243 с.**

**15. Биллиг В.А., Мусикаев И.Х. «Visual C++ 4. Книга для программистов». - М.: Издательский отдел «Русская редакция» ТОО «Channel Trading Ltd.» , 1996. - 352 с. ил.**

**16. Web Database Application with PHP and MySQL, 2nd Edition By David Lane, Hugh E. Williams. © O'Reilly, May 2004. ISBN: 0-596-00543-1.**

**17.CMS List. Обзор cms. Сайт о системах управления сайтом. <http://www.cmslist.ru>**

**18.Joomla! CMS по-русски. <http://joomlaportal.ru>**

**19.Материал из Википедии — свободной энциклопедии о системах управления сайтом. <http://ru.wikipedia.org/wiki/CMS>**

**20.Content management system <http://www.brutto.ru/informacija/uznat-bolshe/content-management-system>**

**21.Joomla-инструменты. <http://docs.joom.ru/!/;C651=00:Categories>**

**22.CMS обзор: CMS, движок Базы данных, система управления сайтом, mambo, php nuke, netcat, phpbb, invision power board, vbulletin. <http://cmsobzor.ru/news.php>**

- 23. Borland InterBase Workgroup Server. API Guide. - Borland International Inc, 1995 - 330 с.**
- 24. Borland InterBase Workgroup Server. DataDefinition Guide. - Borland International Inc, 1995 - 212 с.**
- 25. Borland InterBase Workgroup Server. Language Reference. - Borland International Inc, 1995 - 234 с.**
- 26. Borland InterBase Workgroup Server. Programmer's Guide. - Borland International Inc, 1995 - 340 с.**
- 27. Microsoft Online Documentation: Win32 Programmers Reference.**
- 14 R.Barker "CASE\* Method - Entity Relationship Modelling". - Oracle Inc., 1990 - 243 с.**
- 28. Биллиг В.А., Мусикаев И.Х. «Visual C++ 4. Книга для программистов». - М.: Издательский отдел «Русская редакция» ТОО «Channel Trading Ltd.» , 1996. - 352 с. ил.**

## ПРИЛОЖЕНИЯ

### ИСХОДНЫЙ КОД ПРОГРАММЫ

```
<? include ("lock.php");
//include("blocks/bd.php");
@mysql_query("SET NAMES 'UTF8'");
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<link rel="stylesheet" type="text/css" href="../images/styles.css" />
<title>Военкомат: сведения о призывниках</title>
</head>
<body>
<div id="wrap">
<div id="title"></div>
<div id="menu">
<ul id="lmenu">
<p>Редактирование</p>
<li><a href="add_all.php">Добавить &raquo;</a> </li>
<li><a href="change_data_list.php">Изменить &raquo;</a> </li>
<li><a href="delete_data.php">Удалить &raquo;</a></li>
</ul>
```

```

<? include("blocks/pmenu.php") ?>
</div>
<p id="text">
<style type="text/css">
h1,h2 {color:#CA176C; text-decoration:underline; text-align:center;}
p {padding:0; margin:0 10px 0 0 ; color:#900; font:bold 14px "Times New Roman", Times, serif; }
#wrapn{
min-height:350px;
margin:5px;
float:left;
width: 740px;
border:1px solid #424242;
padding:10px;}
#wrapn p{padding:15px; font-size:18px;}
#wrapn p a{color:#CA176C;}
#wrapn p a:hover{text-decoration:none;}
input {border:1px solid #900; border-radius:3px; box-shadow:2px 2px 3px #999999; }
#submit{position:relative; left:0; float:right;}
</style>
<? //include ("lock.php")?>
<div id="wrapn">
<h1 >Главная страница отдела по дела обороны г. Ферганы</h1>

```

**<h2>Выберите призывника для просмотра его данных</h2>**

**<?>**

```
@mysql_query("SET names 'UTF8'");
```

```
$result = mysql_query("SELECT Uch_nomer,Fam, Imya, Otch, id  
FROM Lich_dannie ORDER BY id ASC",$db);
```

```
$myrow = mysql_fetch_array($result);
```

```
do
```

```
{
```

```
printf ("<p><a href='show_all_data.php?id=%d'>&raquo;
```

```
Учетный № %s: %s %s
```

```
%s</a></p>",$myrow["id"],$myrow["Uch_nomer"],$myrow["Fa  
m"],$myrow["Imya"],$myrow["Otch"]);
```

```
}
```

```
while ($myrow = mysql_fetch_array($result));
```

**?>**

**</div>**

**</p>**

**<div id="footer">&reg; Отдел по делам обороны г.  
Ферганы</div>**

**</div>**

**</body>**

**</html>**

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<link rel="stylesheet" type="text/css" href="images/styles.css" />
<title>Военкомат: сведения о призывниках</title>
</head>
<body>
<div id="wrap">
<div id="title"></div>
<div id="menu">
<ul id="lmenu">
<p >Редактирование</p>
<li><a href="index.php"> Главная &raquo;</a> </li>
<li><a href="add_all.php">Добавить &raquo;</a> </li>
<li><a href="change_data_list.php">Изменить &raquo;</a> </li>
<li><a href="delete_data.php">Удалить &raquo;</a></li>
</ul>
<? include("blocks/pmenu.php") ?>
</div>
<style type="text/css">
h1,h2 {color:#CA176C; text-decoration:underline; text-
align:center;}

```

```

p {padding:0; margin:0 10px 0 0 ; color:#900; font:bold 14px
"Times New Roman", Times, serif; }
#wrapn{
margin:5px;
float:left;
width: 740px;
border:1px solid #424242;
padding:10px;}
#wrapn p{float:left;}
input {position: absolute; left:450px; border:1px solid #900; border-
radius:3px; box-shadow:2px 2px 3px #999999; }
#submit{position:relative; left:0; float:right;}
</style>
<body>
<div id="wrapn">
<h1>Учетная карточка призывника</h1>
<h2>Ниже введите личные данные призывника в поля (*
обязательные поля)</h2>
<form      action="add/add_all_data.php"      method="post"
name="form1">
<p>Учетный номер призывника:</p>
<input      name="Uch_nomer"      type="text"      size="30"
maxlength="30" />
<br/> <br/>
<p>Фамилия: </p>
<input name="Fam" type="text" size="30" maxlength="30" />

```

```
<br/> <br/>
<p>Имя: </p>
<input name="Имя" type="text" size="30" maxlength="30" />
<br/> <br/>
<p>Отчество: </p>
<input name="Otch" type="text" size="30" maxlength="30" />
<br/> <br/>
<p>Серия паспорта: </p>
<input name="Seria_pass" type="text" size="10" maxlength="10"
/>
<br/> <br/>
<p>Номер паспорта: </p>
<input name="N_pass" type="text" size="20" maxlength="20" />
<br/> <br/>
<p>Дата выдачи: </p>
<input name="Data_vid_pass" type="text" size="20"
maxlength="20" />
<br/> <br/>
<p>Кем выдан: </p>
<input name="Kem_vid_pass" type="text" size="30"
maxlength="30" />
<br/> <br/>
<p>Фото: </p>
<input name="Foto" type="file" />
<br/> <br/>
<h1>Общие сведения о призывнике</h1>
```

**<p>Дата рождения:</p>**

**<input name="data\_rojd" type="text" size="30" maxlength="30" />**

**<br/> <br/>**

**<p>Место рождения: </p>**

**<input name="mesto\_rojd" type="text" size="30" maxlength="30" />**

**<br/> <br/>**

**<p>Национальность: </p>**

**<input name="Nacionalnost" type="text" size="30" maxlength="30" />**

**<br/> <br/>**

**<p>Адрес места жительства: </p>**

**<input name="Adres\_mesta\_jit" type="text" size="30" maxlength="30" />**

**<br/> <br/>**

**<p>Социальное происхождения: </p>**

**<input name="Soc\_proishoj" type="text" size="30" maxlength="50" />**

**<br/> <br/>**

**<p>Родной язык: </p>**

**<input name="Rodnoi\_yazik" type="text" size="20" maxlength="20" />**

**<br/> <br/>**

**<p>Владение государственным <br /> языком: </p>**

```
<input      name="Vlad_gos_yaz"      type="text"      size="20"
maxlength="20" />
<br/> <br/>
<p>Владение иностранным языком: </p>
<input      name="Vlad_inostr_yaz"    type="text"      size="30"
maxlength="30" />
<br/> <br/>
<p>Место учебы:</p>
<input name="Mesto_uch" type="text" size="30" maxlength="50"
/>
<br/> <br/>
<p>Срок обучения: </p>
<input      name="Srok_obuch"        type="text"      size="10"
maxlength="20" />
<br/> <br/>
<p>Номер курса: </p>
<input      name="Nomer_kursa"      type="text"      size="10"
maxlength="20" />
<br/> <br/>
<p>Дата окончания учебы: </p>
<input      name="Data_okonch_uch"   type="text"      size="30"
maxlength="30" />
<br/> <br/>
<p>Специальность:</p>
<input      name="Specialnost"      type="text"      size="30"
maxlength="30" />
```

<br/> <br/>

<p>Образование: </p>

<input name="Образование" type="text" size="30" maxlength="30" />

<br/> <br/>

<p>Место работы: </p>

<input name="Mesto\_rab" type="text" size="30" maxlength="30" />

<br/> <br/>

<p>Гражданская профессия: </p>

<input name="Grajd\_prof" type="text" size="30" maxlength="30" />

<br/> <br/>

<p>Вид спорта: </p>

<input name="Vid\_sporta" type="text" size="30" maxlength="10" />

<br/> <br/>

<p>Разряд по спорту: </p>

<input name="Razr\_po\_sportu" type="text" size="10" maxlength="20" />

<br/> <br/>

<p>С каким государством <br /> имеет связь: </p>

<input name="S\_kakim\_gos\_im\_svyaz" type="text" size="30" maxlength="20" />

<br/> <br/>

<p>С каким гос-вом имеют <br />связь родственники: </p>

```
<input name="S_kakim_gos_im_svyaz_rodstv" type="text"
size="30" maxlength="30" />
<br/> <br/>
<p>Судимость:</p>
<input name="Sudimost" type="text" size="30" maxlength="30"
/>
<br/> <br/>
<p>Судимость родственников: </p>
<input name="Sudimost_rodstv" type="text" size="30"
maxlength="20" />
<br/> <br/>
<h1>Состав семьи призывника</h1>
<p>Родство:</p>
<input name="Rodstvo" type="text" size="30" maxlength="50" />
<br/> <br/>
<p>Фамилия имя и отчество: </p>
<input name="FIO" type="text" size="30" maxlength="50" />
<br/> <br/>
<p>Год рождения: </p>
<input name="God_rojd" type="text" size="30" maxlength="30"
/>
<br/> <br/>
<p>Домашний адрес: </p>
<input name="Dom_adres" type="text" size="30"
maxlength="50" />
<br/> <br/>
```

```
<p>Место работы: </p>
<input      name="Mesto_rab_s"      type="text"      size="30"
maxlength="50" />
<br/> <br/>
<h1>Результаты приписки к призывному участку</h1>
<p>Рост:</p>
<input name="Rost" type="text" size="30" maxlength="50" />
<br/> <br/>
<p>Вес: </p>
<input name="Ves" type="text" size="30" maxlength="50" />
<br/> <br/>
<p>Объем груди: </p>
<input name="Ob_grudi" type="text" size="30" maxlength="30"
/>
<br/> <br/>
<p>Спирометрия: </p>
<input      name="Spirometriya"      type="text"      size="30"
maxlength="50" />
<br/> <br/>
<p>Степень ограничения: </p>
<input  name="Stepen_ogranicheniya"  type="text"  size="30"
maxlength="50" />
<br/> <br/>
<h2>Заключение врачей - специалистов</h2>
<p>Хирурга:</p>
<br />
```

Степень годности:

<br/><br/>

Степень ограничения:

<br/> <br/>

<p>Терапевта:</p>

<br />

Степень годности:

<br/><br/>

Степень ограничения:

<br/> <br/>

<p>Данные флюорографии:</p>

<br />

Степень годности:

<br/><br/>

Степень ограничения:

<br/> <br/>

<p>Невропатолог:</p>

<br />

Степень годности:

<br/><br/>

Степень ограничения: <input name="Nevropatolog\_so" type="text" size="30" maxlength="50" />

<br/> <br/>

<p>Психиатр:</p>

<br />

Степень годности: <input name="Psihiatr\_sg" type="text" size="30" maxlength="50" />

<br/><br/>

Степень ограничения: <input name="Psihiatr\_so" type="text" size="30" maxlength="50" />

<br/> <br/>

<p>Окулист:</p>

<br />

Степень годности: <input name="Okulist\_sg" type="text" size="30" maxlength="50" />

<br/><br/>

Степень ограничения: <input name="Okulist\_so" type="text" size="30" maxlength="50" />

<br/> <br/>

<p>Отоларинголог:</p>

<br />

Степень годности: <input name="Otolaringolog\_sg" type="text" size="30" maxlength="50" />

<br/><br/>

Степень ограничения:

<br/> <br/>

<p>Стоматолог:</p>

<br />

Степень годности:

<br/><br/>

Степень ограничения:

<br/> <br/>

<p>Годен к военной службе:</p>

<br />

Степень годности:

<br/><br/>

Степень ограничения:

<br/> <br/>

<p>Годен к военной службе в ВДВ и погранвойсках негоден:</p>

<br />

Степень годности:

<br/><br/>

Степень ограничения:

<br/> <br/>

<p>Не годен к военной службе, по статье:</p>

<br />

Степень годности:

<br/><br/>

Степень ограничения:

<br/> <br/>

<p>Отсрочка на лечение:</p>

<br />

Степень годности:

<br/><br/>

Степень ограничения:

<br/> <br/>

<p>Негоден к военной службе в мирное время, годен к нестроевой в военное время по статье:</p>

<br />

Степень годности:

<br/><br/>

Степень ограничения:

<br/> <br/>

<p>Негоден к военной службе с исключением с воинского учета по статье:</p>

<br />

Степень годности:

<br/><br/>

Степень ограничения:

<br/> <br/>

<p>Нуждается в отсрочке на основании:</p>

<br />

Степень годности:

<br/><br/>

Степень ограничения:

<br/> <br/>

<p>Предназначен:</p>

<br />

Степень годности:

<br/><br/>

Степень ограничения:

<br/> <br/>

<h1>Подготовка к службе в вооруженных силах</h1>

<p>Отметка о получении <br />начальной военной подготовки:</p>

<input name="Poluch\_NVP" type="text" size="30" maxlength="50" />

<br/> <br/>

<p>Заключение о профпригодности: </p>

<input name="Zakl\_profprigod" type="text" size="30" maxlength="50" />

<br/> <br/>

<p>Учебное учреждение <br />(организация ОСО): </p>

<input name="Uch\_uchrejd" type="text" size="30" maxlength="30" />

<br/> <br/>

<p>Получаемая специальность<br /> в учебном учреждении: </p>

<input name="Specialnost\_p" type="text" size="30" maxlength="50" />

<br/> <br/>

<p>Дата направления на учебу: </p>

<input name="Data\_napr\_na\_uch" type="text" size="30" maxlength="50" />

<br/> <br/>

**<p>Отметка о начале учебы (дата): </p>**

**<input name="Nachalo\_uch" type="text" size="30" maxlength="50" />**

**<br/> <br/>**

**<p>Срок обучения: </p>**

**<input name="Srok\_obuch\_p" type="text" size="30" maxlength="50" />**

**<br/> <br/>**

**<p>Отметка об окончании <br />учебы (дата): </p>**

**<input name="Okonch\_uch" type="text" size="30" maxlength="50" />**

**<br/> <br/>**

**<h1>Явка в районный отдел по делам обороны для отправки в воинскую часть</h1>**

**<p>Дата отправки:</p>**

**<input name="Data\_otpr" type="text" size="30" maxlength="50" />**

**<br/> <br/>**

**<p>Номер воинской <br />команды (К-ОШВС): </p>**

**<input name="N\_komandi" type="text" size="30" maxlength="50" />**

**<br/> <br/>**

**<p>Военный округ, воинская часть: </p>**

**<input name="Voen\_okrug" type="text" size="30" maxlength="30" />**

**<br/> <br/>**

**<p>Причина неявки в районный<br /> отдел по делам обороны:  
</p>**

**<input name="Prichina\_neyavki" type="text" size="30"  
maxlength="50" />**

**<br/> <br/>**

**<p>Причина возврата с<br /> областного сборного пункта: </p>**

**<input name="Prich\_vozvr\_s\_oblpunkta" type="text" size="30"  
maxlength="50" />**

**<br/> <br/>**

**<h1>Прием на воинский учет и снятие с воинского учета</h1>**

**<p>Принят на воинский учет:</p>**

**<input name="Prinyat" type="text" size="30" maxlength="50" />**

**<br/> <br/>**

**<p>Снят с воинского учета: </p>**

**<input name="Snyat" type="text" size="30" maxlength="50" />**

**<br/> <br/>**

**<input name="submit" type="submit" id="submit"  
value="Сохранить данные" />**

**<br />**

**</form>**

**</div>**

**</p>**

**<div id="footer">&reg; Отдел по делам обороны г.  
Ферганы</div>**

**</div>**

**</body>**

**</html>**

<?

```
include ('blocks/bd.php');
@mysql_query('SET names 'UTF8');
$id = $_GET['id'];
$result1 = mysql_query ("SELECT * FROM Lich_dannie WHERE
id = $id",$db);
$myrow = mysql_fetch_array($result1);
$Uch_nomer = $myrow['Uch_nomer'];
$Fam = $myrow['Fam'];
$Imya = $myrow['Imya'];
$Otch = $myrow['Otch'];
$Seria_pass = $myrow['Seria_pass'];
$N_pass = $myrow['N_pass'];
$Data_vid_pass = $myrow['Data_vid_pass'];
$Kem_vid_pass = $myrow['Kem_vid_pass'];
$Foto = $myrow['Foto'];
$result2 = mysql_query ("SELECT * FROM Obshie_dannie
WHERE id_prizivnika = $id",$db);
$myrow1 = mysql_fetch_array($result2);
$data_rojd = $myrow1['data_rojd'];
$mesto_rojd = $myrow1['mesto_rojd'];
$Nacionalnost = $myrow1['Nacionalnost'];
$Adres_mesta_jit = $myrow1['Adres_mesta_jit'];
$Soc_proishoj = $myrow1['Soc_proishoj'];
$Rodnoi_yazik = $myrow1['Rodnoi_yazik'];
$Vlad_gos_yaz = $myrow1['Vlad_gos_yaz'];
```

```

$Vlad_inostr_yaz = $myrow1['Vlad_inostr_yaz'];
$Mesto_uch = $myrow1['Mesto_uch'];
$Srok_obuch = $myrow1['Srok_obuch'];
$Nomer_kursa = $myrow1['Nomer_kursa'];
$Data_okonch_uch = $myrow1['Data_okonch_uch'];
$Specialnost = $myrow1['Specialnost'];
$Obrazovanie = $myrow1['Obrazovanie'];
$Mesto_rab = $myrow1['Mesto_rab'];
$Grajd_prof = $myrow1['Grajd_prof'];
$Vid_sporta = $myrow1['Vid_sporta'];
$Razr_po_sportu = $myrow1['Razr_po_sportu'];
$S_kakim_gos_im_svyaz = $myrow1['S_kakim_gos_im_svyaz'];
$S_kakim_gos_im_svyaz_rodstv =
$myrow1['S_kakim_gos_im_svyaz_rodstv'];
$Sudimost = $myrow1['Sudimost'];
$Sudimost_rodstv = $myrow1['Sudimost_rodstv'];
$result3 = mysql_query ('SELECT * FROM Sostav_semii WHERE
id_prizivnika = $id',$db);
$myrow2 = mysql_fetch_array($result3);
$Rodstvo = $myrow2['Rodstvo'];
$FIO = $myrow2['FIO'];
$God_rojd = $myrow2['God_rojd'];
$Dom_adres = $myrow2['Dom_adres'];
$Mesto_rab_s = $myrow2['Mesto_rab'];
$result4 = mysql_query ('SELECT * FROM Rezultati_pripiski
WHERE id_prizivnika = $id',$db);

```

```

$myrow3 = mysql_fetch_array($result4);
$Rost = $myrow3['Rost'];
$Ves = $myrow3['Ves'];
$Ob_grudi = $myrow3['Ob_grudi'];
$Spirometriya = $myrow3['Spirometriya'];
$Stepen_ogranicheniya = $myrow3['Stepen_ogranicheniya'];

$result5 = mysql_query ("SELECT * FROM Zakluch_vrachei
WHERE id_prizivnika = $id", $db);
$myrow4 = mysql_fetch_array($result5);
$Hirurg_sg = $myrow4['Hirurg'];
$Terapevt_sg = $myrow4['Terapevt'];
$Fluorografiya_sg = $myrow4['Fluorografiya'];
$Nevropatolog_sg = $myrow4['Nevropatolog'];
$Psihiatr_sg = $myrow4['Psihiatr'];
$Okulist_sg = $myrow4['Okulist'];
$Otolaringolog_sg = $myrow4['Otolaringolog'];
$Stomatolog_sg = $myrow4['Stomatolog'];
$Goden_k_slujbe_sg = $myrow4['Goden_k_slujbe'];
$V_VDV_i_pogran_ne_goden_sg =
$myrow4['V_VDV_i_pogran_ne_goden'];
$Negoden_po_statie_sg = $myrow4['Negoden_po_statie'];
$Otsrochka_sg = $myrow4['Otsrochka'];
$Negoden_v_mirnoe_sg = $myrow4['Negoden_v_mirnoe'];
$Negoden_s_iskl_ucheta_sg = $myrow4['Negoden_s_iskl_ucheta'];
$Nujd_v_otsrochke_sg = $myrow4['Nujd_v_otsrochke'];

```

```

$Prednaznachen_sg = $myrow4['Prednaznachen'];
$myrow4 = mysql_fetch_array($result5);
$Hirurg_so = $myrow4['Hirurg'];
$Terapevt_so = $myrow4['Terapevt'];
$Fluorografiya_so = $myrow4['Fluorografiya'];
$Nevropatolog_so = $myrow4['Nevropatolog'];
$Psihiatr_so = $myrow4['Psihiatr'];
$Okulist_so = $myrow4['Okulist'];
$Otolaringolog_so = $myrow4['Otolaringolog'];
$Stomatolog_so = $myrow4['Stomatolog'];
$Goden_k_slujbe_so = $myrow4['Goden_k_slujbe'];
$V_VDV_i_pogran_ne_goden_so =
$myrow4['V_VDV_i_pogran_ne_goden'];
$Negoden_po_statie_so = $myrow4['Negoden_po_statie'];
$Otsrochka_so = $myrow4['Otsrochka'];
$Negoden_v_mirnoe_so = $myrow4['Negoden_v_mirnoe'];
$Negoden_s_iskl_ucheta_so = $myrow4['Negoden_s_iskl_ucheta'];
$Nujd_v_otsrochke_so = $myrow4['Nujd_v_otsrochke'];
$Prednaznachen_so = $myrow4['Prednaznachen'];
$result6 = mysql_query ('SELECT * FROM Podgotovka_k_sluj
WHERE id_prizivnika = $id',$db);
$myrow5 = mysql_fetch_array($result6);
$Poluch_NVP = $myrow5['Poluch_NVP'];
$Zakl_profprigod = $myrow5['Zakl_profprigod'];
$Uch_uchrejd = $myrow5['Uch_uchrejd'];
$Specialnost_p = $myrow5['Specialnost'];

```

```

$Data_napr_na_uch = $myrow5['Data_napr_na_uch'];
$Nachalo_uch = $myrow5['Nachalo_uch'];
$Srok_obuch_p = $myrow5['Srok_obuch'];
$Okonch_uch = $myrow5['Okonch_uch'];
$result7 = mysql_query ("SELECT * FROM Yavka_v_raiotdel
WHERE id_prizivnika = $id",$db);
$myrow6 = mysql_fetch_array($result7);
$Data_otpr = $myrow6['Data_otpr'];
$N_komandi = $myrow6['N_komandi'];
$Voen_okrug = $myrow6['Voen_okrug'];
$Prichina_neyavki = $myrow6['Prichina_neyavki'];
$Prich_vozvr_s_oblpunkta = $myrow6['Prich_vozvr_s_oblpunkta'];
$result8 = mysql_query ("SELECT * FROM
Priem_i_snyat_s_ucheta WHERE id_prizivnika = $id",$db);
$myrow7 = mysql_fetch_array($result8);
$Prinyat = $myrow7['Prinyat'];
$Snyat = $myrow7['Snyat'];
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<link rel="stylesheet" type="text/css" href="images/styles.css" />

```

```

<title>Военкомат: сведения о призывниках</title>
</head>
<body>
<div id="wrap">
<div id="title"></div>
<div id="menu">
<ul id="lmenu">
<p >Редактирование</p>
<li><a href="index.php"> Главная &raquo;</a> </li>
<li><a href="add_all.php">Добавить &raquo;</a> </li>
<li><a href="change_data_list.php">Изменить &raquo;</a> </li>
<li><a href="delete_data.php">Удалить &raquo;</a></li>
</ul>
<? include("blocks/pmenu.php") ?>
</div>
<style type="text/css">
h1,h2 {color:#CA176C; text-decoration:underline; text-align:center;}
p {padding:0; margin:0 10px 0 0 ; color:#900; font:bold 14px "Times New Roman", Times, serif; }
#wrapn{
margin:5px;
float:left;
width: 740px;
border:1px solid #424242;
padding:10px;}

```

```

#wrapn p{float:left;}
input {position: absolute; left:450px; border:1px solid #900; border-
radius:3px; box-shadow:2px 2px 3px #999999; }
#submit{position:relative; left:0; float:right;}
</style>
<div id="wrapn">
<h1>Изменение учетных данных призывника</h1>
<h2>Ниже введите личные данные призывника в поля (*
обязательные поля)</h2>
<form      action="add/update_all_data.php"      method="post"
name="form1">
<input name="id" type="hidden" value="<? echo $id ?>" />
<p>Учетный номер призывника:</p>
<input      name="Uch_nomer"      type="text"      size="30"
maxlength="30" value="<? echo $Uch_nomer ?>" />
<br/> <br/>
<p>Фамилия: </p>
<input name="Fam" type="text" size="30" maxlength="30"
value="<? echo $Fam ?>" />
<br/> <br/>
<p>Имя: </p>
<input name="Imya" type="text" size="30" maxlength="30"
value="<? echo $Imya ?>" />
<br/> <br/>
<p>Отчество: </p>

```

```
<input name="Otch" type="text" size="30" maxlength="30"
value="<? echo $Otch ?>"/>
<br/> <br/>
<p>Серия паспорта: </p>
<input name="Seria_pass" type="text" size="10" maxlength="10"
value="<? echo $Seria_pass ?>"/>
<br/> <br/>
<p>Номер паспорта: </p>
<input name="N_pass" type="text" size="20" maxlength="20"
value="<? echo $N_pass ?>"/>
<br/> <br/>
<p>Дата выдачи: </p>
<input name="Data_vid_pass" type="text" size="20"
maxlength="20" value="<? echo $Data_vid_pass ?>"/>
<br/> <br/>
<p>Кем выдан: </p>
<input name="Kem_vid_pass" type="text" size="30"
maxlength="30" value="<? echo $Kem_vid_pass ?>"/>
<br/> <br/>
<p>Фото: </p>
<input name="Foto" type="file" src="<? echo $Foto ?>" />
<br/> <br/>
<h1>Общие сведения о призывнике</h1>
<p>Дата рождения:</p>
<input name="data_rojd" type="text" size="30" maxlength="50"
value="<? echo $data_rojd ?> "/>
```

<br/> <br/>

<p>Место рождения: </p>

<input name="mesto\_rojd" type="text" size="30" maxlength="50" value="<? echo \$mesto\_rojd ?>"/>

<br/> <br/>

<p>Национальность: </p>

<input name="Nacionalnost" type="text" size="30" maxlength="50" value="<? echo \$Nacionalnost ?>"/>

<br/> <br/>

<p>Адрес места жительства: </p>

<input name="Adres\_mesta\_jit" type="text" size="30" maxlength="50" value="<? echo \$Adres\_mesta\_jit ?>"/>

<br/> <br/>

<p>Социальное происхождения: </p>

<input name="Soc\_proishoj" type="text" size="30" maxlength="50" value="<? echo \$Soc\_proishoj ?>"/>

<br/> <br/>

<p>Родной язык: </p>

<input name="Rodnoi\_yazik" type="text" size="20" maxlength="20" value="<? echo \$Rodnoi\_yazik ?>"/>

<br/> <br/>

<p>Владение государственным <br /> языком: </p>

<input name="Vlad\_gos\_yaz" type="text" size="20" maxlength="20" value="<? echo \$Vlad\_gos\_yaz ?>"/>

<br/> <br/>

<p>Владение иностранным языком: </p>

```
<input name="Vlad_inostr_yaz" type="text" size="30"
maxlength="50" value="<? echo $Vlad_inostr_yaz ?>"/>
<br/> <br/>
<p>Место учебы:</p>
<input name="Mesto_uch" type="text" size="30" maxlength="50"
value="<? echo $Mesto_uch ?>"/>
<br/> <br/>
<p>Срок обучения: </p>
<input name="Srok_obuch" type="text" size="10"
maxlength="20" value="<? echo $Srok_obuch ?>"/>
<br/> <br/>
<p>Номер курса: </p>
<input name="Nomer_kursa" type="text" size="10"
maxlength="20" value="<? echo $Nomer_kursa ?>"/>
<br/> <br/>
<p>Дата окончания учебы: </p>
<input name="Data_okonch_uch" type="text" size="30"
maxlength="30" value="<? echo $Data_okonch_uch ?>"/>
<br/> <br/>
<p>Специальность:</p>
<input name="Specialnost" type="text" size="30"
maxlength="50" value="<? echo $Specialnost ?>"/>
<br/> <br/>
<p>Образование: </p>
<input name="Obrazovanie" type="text" size="30"
maxlength="50" value="<? echo $Obrazovanie ?>"/>
```

<br/> <br/>

<p>Место работы: </p>

<input name="Mesto\_rab" type="text" size="30" maxlength="50" value="<? echo \$Mesto\_rab ?>"/>

<br/> <br/>

<p>Гражданская профессия: </p>

<input name="Grajd\_prof" type="text" size="30" maxlength="50" value="<? echo \$Grajd\_prof ?>"/>

<br/> <br/>

<p>Вид спорта: </p>

<input name="Vid\_sporta" type="text" size="30" maxlength="50" value="<? echo \$Vid\_sporta ?>"/>

<br/> <br/>

<p>Разряд по спорту: </p>

<input name="Razr\_po\_sportu" type="text" size="10" maxlength="20" value="<? echo \$Razr\_po\_sportu ?>"/>

<br/> <br/>

<p>С каким государством <br /> имеет связь: </p>

<input name="S\_kakim\_gos\_im\_svyaz" type="text" size="30" maxlength="50" value="<? echo \$S\_kakim\_gos\_im\_svyaz ?>"/>

<br/> <br/>

<p>С каким гос-вом имеют <br />связь родственники: </p>

<input name="S\_kakim\_gos\_im\_svyaz\_rodstv" type="text" size="30" maxlength="50" value="<? echo \$S\_kakim\_gos\_im\_svyaz\_rodstv ?>"/>

<br/> <br/>

```
<p>Судимость:</p>
<input name="Sudimost" type="text" size="30" maxlength="50"
value="<? echo $Sudimost ?>"/>
<br/> <br/>
<p>Судимость родственников: </p>
<input name="Sudimost_rodstv" type="text" size="30"
maxlength="50" value="<? echo $Sudimost_rodstv ?>"/>
<br/> <br/>
<h1>Состав семьи призывника</h1>
<p>Родство:</p>
<input name="Rodstvo" type="text" size="30" maxlength="50"
value="<? echo $Rodstvo ?>"/>
<br/> <br/>
<p>Фамилия имя и отчество: </p>
<input name="FIO" type="text" size="30" maxlength="50"
value="<? echo $FIO ?>"/>
<br/> <br/>
<p>Год рождения: </p>
<input name="God_rojd" type="text" size="30" maxlength="50"
value="<? echo $God_rojd ?>"/>
<br/> <br/>
<p>Домашний адрес: </p>
<input name="Dom_adres" type="text" size="30"
maxlength="50" value="<? echo $Dom_adres ?>"/>
<br/> <br/>
<p>Место работы: </p>
```

```
<input      name="Mesto_rab_s"      type="text"      size="30"
maxlength="50" value="<? echo $Mesto_rab_s ?>"/>
<br/> <br/>
<h1>Результаты приписки к призывному участку</h1>
<p>Рост:</p>
<input  name="Rost"  type="text"  size="30"  maxlength="50"
value="<? echo $Rost ?>"/>
<br/> <br/>
<p>Вес: </p>
<input  name="Ves"  type="text"  size="30"  maxlength="50"
value="<? echo $Ves ?>"/>
<br/> <br/>
<p>Объем груди: </p>
<input name="Ob_grudi" type="text" size="30" maxlength="50"
value="<? echo $Ob_grudi ?>"/>
<br/> <br/>
<p>Спирометрия: </p>
<input      name="Spirometriya"      type="text"      size="30"
maxlength="50" value="<? echo $Spirometriya ?>"/>
<br/> <br/>
<p>Степень ограничения: </p>
<input  name="Stepen_ogranicheniya"  type="text"  size="30"
maxlength="50" value="<? echo $Stepen_ogranicheniya ?>"/>
<br/> <br/>
<h2>Заключение врачей - специалистов</h2>
<p>Хирурга:</p>
```

<br />

Степень годности: <input name="Hirurg\_sg" type="text" size="30" maxlength="50" value="<? echo \$Hirurg\_sg ?>"/>

<br/><br/>

Степень ограничения: <input name="Hirurg\_so" type="text" size="30" maxlength="50" value="<? echo \$Hirurg\_so ?>"/>

<br/> <br/>

<p>Терапевта:</p>

<br />

Степень годности: <input name="Terapevt\_sg" type="text" size="30" maxlength="50" value="<? echo \$Terapevt\_sg ?>"/>

<br/><br/>

Степень ограничения: <input name="Terapevt\_so" type="text" size="30" maxlength="50" value="<? echo \$Terapevt\_so ?>"/>

<br/> <br/>

<p>Данные флюорографии:</p>

<br />

Степень годности: <input name="Fluorografiya\_sg" type="text" size="30" maxlength="50" value="<? echo \$Fluorografiya\_sg ?>"/>

<br/><br/>

Степень ограничения: <input name="Fluorografiya\_so" type="text" size="30" maxlength="50" value="<? echo \$Fluorografiya\_so ?>"/>

<br/> <br/>

<p>Невропатолог:</p>

<br />

Степень годности: <input name="Nevropatolog\_sg" type="text" size="30" maxlength="50" value="<? echo \$Nevropatolog\_sg ?>"/>

<br/><br/>

Степень ограничения: <input name="Nevropatolog\_so" type="text" size="30" maxlength="50" value="<? echo \$Nevropatolog\_so ?>"/>

<br/> <br/>

<p>Психиатр:</p>

<br />

Степень годности: <input name="Psihiatr\_sg" type="text" size="30" maxlength="50" value="<? echo \$Psihiatr\_sg ?>"/>

<br/><br/>

Степень ограничения: <input name="Psihiatr\_so" type="text" size="30" maxlength="50" value="<? echo \$Psihiatr\_so ?>"/>

<br/> <br/>

<p>Окулист:</p>

<br />

Степень годности: <input name="Okulist\_sg" type="text" size="30" maxlength="50" value="<? echo \$Okulist\_sg ?>"/>

<br/><br/>

Степень ограничения: <input name="Okulist\_so" type="text" size="30" maxlength="50" value="<? echo \$Okulist\_so ?>"/>

<br/> <br/>

<p>Отоларинголог:</p>

<br />

Степень годности:

<br/><br/>

Степень ограничения:

<br/> <br/>

<p>Стоматолог:</p>

<br />

Степень годности:

<br/><br/>

Степень ограничения:

<br/> <br/>

<p>Годен к военной службе:</p>

<br />

Степень годности:

<br/><br/>

Степень ограничения:

<br/> <br/>

<p>Годен к военной службе в ВДВ и погранвойсках  
негоден:</p>

<br />

Степень годности: <input  
name="V\_VDV\_i\_pogran\_ne\_goden\_sg" type="text" size="30"  
maxlength="50" value="<? echo \$V\_VDV\_i\_pogran\_ne\_goden\_sg  
?>"/>

<br/><br/>

Степень ограничения: <input  
name="V\_VDV\_i\_pogran\_ne\_goden\_so" type="text" size="30"  
maxlength="50" value="<? echo \$V\_VDV\_i\_pogran\_ne\_goden\_so  
?>"/>

<br/> <br/>

<p>Не годен к военной службе, по статье:</p>

<br />

Степень годности: <input name="Negoden\_po\_statie\_sg"  
type="text" size="30" maxlength="50" value="<? echo  
\$Negoden\_po\_statie\_sg ?>"/>

<br/><br/>

Степень ограничения: <input name="Negoden\_po\_statie\_so"  
type="text" size="30" maxlength="50" value="<? echo  
\$Negoden\_po\_statie\_so ?>"/>

<br/> <br/>

<p>Отсрочка на лечение:</p>

<br />

Степень годности: <br/><br/>

Степень ограничения: <br/> <br/>

<p>Негоден к военной службе в мирное время, годен к нестроевой в военное время по статье:</p>  
<br />

Степень годности: <br/><br/>

Степень ограничения: <br/> <br/>

<p>Негоден к военной службе с исключением с воинского учета по статье:</p>  
<br />

Степень годности: <br/><br/>

Степень ограничения:

<br/> <br/>

<p>Нуждается в отсрочке на основании:</p>

<br />

Степень годности:

<br/><br/>

Степень ограничения:

<br/> <br/>

<p>Предназначен:</p>

<br />

Степень годности:

<br/><br/>

Степень ограничения:

<br/> <br/>

<h1>Подготовка к службе в вооруженных силах</h1>

<p>Отметка о получении <br />начальной военной подготовки:</p>

<input name="Poluch\_NVP" type="text" size="30" maxlength="50" value="<? echo \$Poluch\_NVP ?>"/>  
<br/> <br/>

<p>Заключение о профпригодности: </p>

<input name="Zakl\_profprigod" type="text" size="30" maxlength="50" value="<? echo \$Zakl\_profprigod ?>"/>  
<br/> <br/>

<p>Учебное учреждение <br />(организация ОО): </p>

<input name="Uch\_uchrejd" type="text" size="30" maxlength="50" value="<? echo \$Uch\_uchrejd ?>"/>  
<br/> <br/>

<p>Получаемая специальность<br /> в учебном учреждении:  
</p>

<input name="Specialnost\_p" type="text" size="30" maxlength="50" value="<? echo \$Specialnost\_p ?>"/>  
<br/> <br/>

<p>Дата направления на учебу: </p>

<input name="Data\_napr\_na\_uch" type="text" size="30" maxlength="50" value="<? echo \$Data\_napr\_na\_uch ?>"/>  
<br/> <br/>

<p>Отметка о начале учебы (дата): </p>

<input name="Nachalo\_uch" type="text" size="30" maxlength="50" value="<? echo \$Nachalo\_uch ?>"/>  
<br/> <br/>

**<p>Срок обучения: </p>**

**<input name="Srok\_obuch\_p" type="text" size="30" maxlength="50" value="<? echo \$Srok\_obuch\_p ?>"/>**

**<br/> <br/>**

**<p>Отметка об окончании <br />учебы (дата): </p>**

**<input name="Okonch\_uch" type="text" size="30" maxlength="50" value="<? echo \$Okonch\_uch ?>"/>**

**<br/> <br/>**

**<h1>Явка в районный отдел по делам обороны для отправки в воинскую часть</h1>**

**<p>Дата отправки:</p>**

**<input name="Data\_otpr" type="text" size="30" maxlength="50" value="<? echo \$Data\_otpr ?>"/>**

**<br/> <br/>**

**<p>Номер воинской <br />команды (К-ОШВС): </p>**

**<input name="N\_komandi" type="text" size="30" maxlength="50" value="<? echo \$N\_komandi ?>"/>**

**<br/> <br/>**

**<p>Военный округ, воинская часть: </p>**

**<input name="Voen\_okrug" type="text" size="30" maxlength="50" value="<? echo \$Voen\_okrug ?>"/>**

**<br/> <br/>**

**<p>Причина неявки в районный<br /> отдел по делам обороны:  
</p>**

**<input name="Prichina\_neyavki" type="text" size="30" maxlength="50" value="<? echo \$Prichina\_neyavki ?>"/>**

```
<br/> <br/>
```

```
<p>Причина возврата с<br /> областного сборного пункта: </p>
```

```
<input name="Prich_vozvr_s_oblpunkta" type="text" size="30"
maxlength="50" value="<? echo $Prich_vozvr_s_oblpunkta ?>"/>
```

```
<br/> <br/>
```

```
<h1>Прием на воинский учет и снятие с воинского учета</h1>
```

```
<p>Принят на воинский учет:</p>
```

```
<input name="Prinyat" type="text" size="30" maxlength="50"
value="<? echo $Prinyat ?>"/>
```

```
<br/> <br/>
```

```
<p>Снят с воинского учета: </p>
```

```
<input name="Snyat" type="text" size="30" maxlength="50"
value="<? echo $Snyat ?>"/>
```

```
<br/> <br/>
```

```
<input name="submit" type="submit" id="submit"
value="Сохранить данные" />
```

```
<br />
```

```
</form>
```

```
</div>
```

```
</p>
```