

**МИНИСТЕРСТВО РАЗВИТИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ  
И КОММУНИКАЦИИ РЕСПУБЛИКИ УЗБЕКИСТАН**

**САМАРКАНДСКИЙ ФИЛИАЛ ТАШКЕНТСКОГО  
УНИВЕРСИТЕТА  
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ  
ИМЕНИ МУХАММАДА АЛ-ХОРАЗМИ**

**ФАКУЛЬТЕТ: «КОМПЬЮТЕРНЫЙ ИНЖИНИРИНГ»**

**КАФЕДРА: «КОМПЬЮТЕРНЫЕ СИСТЕМЫ»**

**ВЫПУСКНАЯ  
КВАЛИФИКАЦИОННАЯ РАБОТА  
«Создание информационной системы с интерактивными услугами для  
пациентов центральной больницы Жамбайского района»**

Рассмотрена и допущена к защите

на заседании кафедры № \_\_\_ от

“ \_\_\_\_\_ ” \_\_\_\_\_ 2018 г.

зав. кафедрой:

\_\_\_\_\_ ст.преп.Абдуллаева Н.И.

“ \_\_\_\_\_ ” \_\_\_\_\_ 2018 г.

Выполнил: студент 4-курса \_\_\_\_\_

Назирбоев Жавохир, 408 гр.

Научный руководитель: \_\_\_\_\_

ст.преп. Абдуллаева Н.И..

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
Глава 1. Проектирования информационных систем	6
1.1. Введение в информационные системы и их классификация	6
1.2. Требования, предъявляемые к информационным системам	9
1.3. Компоненты и структура ИС	
Глава 2. Программные средства для создания информационных систем с интерактивными услугами	19
2.1. Обзор инструментальных средств	19
2.2. Разработка интерактивных сайтов	24
2.3. Проектирование автоматизированных информационных систем	27
Элементы разработанной информационной системы с интерактивными услугами	35
3.1. Основные элементы информационной системы	35
3.2. Основы физиологии труда и комфортные условия жизнедеятельности деятельности человека	42
ЗАКЛЮЧЕНИЕ	47
ЛИТЕРАТУРА	48
Приложение	50

## ВВЕДЕНИЕ

В нынешний век глобализации, когда люди могут свободно общаться посредством новейших технологий, очень важно проводить продуманную государственную политику в данной сфере, имея в виду комплексный подход в дальнейшем развитии нашего общества и интеграции его в мировые информационные связи.

Такие цели поставлены в постановлении Президента Ислама Каримова «О мерах по дальнейшему внедрению и развитию современных информационно-коммуникационных технологий», принятом 21 марта 2012 года. Документ разработан в соответствии с Законом Республики Узбекистан «Об информатизации» и нацелен на повышение эффективности деятельности органов государственного и хозяйственного управления, органов власти на местах, обеспечение широкого использования современных информационно-коммуникационных технологий в сфере государственного и общественного строительства [1].

Постановлением главы государства утверждена также Программа дальнейшего внедрения и развития информационно-коммуникационных технологий в Республике Узбекистан на 2012-2014 годы, в которой подробно и адресно по годам прописаны организационные меры, развитие инфраструктуры ИКТ, подготовка, переподготовка и повышение квалификации кадров, совершенствование правительственного портала, веб-сайтов государственных органов и интерактивных услуг, обеспечение информационной безопасности, применение современных средств ИКТ в деятельности коммерческих структур, популяризация ИКТ среди населения, в том числе в сельской местности [2].

**Актуальность темы.** В сегодняшнее время управление предприятием без компьютера просто немыслимо. Компьютеры давно и прочно вошли в такие области управления, как бухгалтерский учет, управление складом, ассортиментом и закупками. Однако современный бизнес требует гораздо

более широкого применения информационных технологий в управлении предприятием. Жизнеспособность и развитие информационных технологий объясняется тем, что современный бизнес крайне чувствителен к ошибкам в управлении. Интуиции, личного опыта руководителя и размеров капитала уже мало для того, чтобы быть первым. Для принятия любого грамотного управленческого решения в условиях неопределенности и риска необходимо постоянно держать под контролем различные аспекты финансово-хозяйственной деятельности, будь то: торговля, производство или предоставление каких-либо услуг. Поэтому современный подход к управлению предполагает вложение средств в информационные технологии. И чем крупнее предприятие, тем серьезнее должны быть подобные вложения. Они являются жизненной необходимостью — в жесткой конкурентной борьбе одержать победу сможет лишь тот, кто лучше оснащен и наиболее эффективно организован.

Информационная система «Jomboyshifo.uz» включает в себя данные о врачах, пациентах, кабинетах и вызовах, которые необходимы для работы больницы. Актуальность данной темы в том, что в наш век информационных технологий, стало реально все документы преобразовывать в электронный вид и регистратура в считанные минуты может найти сведения о принятых пациентах, вызовах, кабинетах.

**Цель работы.** В данной выпускной квалификационной работе рассматривается разработка модуля интерактивных услуг в информационной системы для пациентов центральной больницы Жамбайского района. В программном модуле разрабатывается автоматизация представление услуг для территориального населения семейной больницы.

**Основные задачи исследования:**

- изучение и анализ проектирование информационных систем в проектирование;
- обзор инструментальных средств для создание информационных систем;

- разработка интерактивный сайт для больницы;
- разработка интерактивные услуги для информационных систем больницы.

**Структура и объем выпускной квалификационной работы.**

Выпускная квалификационная работа состоит из введения, трёх глав и заключения. Содержит список использованной литературы из 22 наименований. В приложение даны код программы.

# ГЛАВА 1. ПРОЕКТИРОВАНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ

## 1.1. Введение в информационные системы и их классификация

Перед рассмотрением конкретных информационных систем дадим несколько необходимых определений:

**Система** (в предметной области) – это множество взаимосвязанных элементов, каждый из которых связан прямо или косвенно с каждым другим элементом, а два любые подмножества этого множества не могут быть независимыми, не нарушая целостность, единство системы.

**Элемент системы** – это простейшая структурная составляющая системы, которая в рамках данной системы не структурируется.

**Структура системы** – это совокупность устойчивых связей, способов взаимодействия элементов системы, определяющая ее целостность и единство.

**Среда** (в предметной области) – это все, что находится в предметной области за границами системы.

Под **информационным процессом** будем понимать процесс, связанный с изменением количества информации в системе в результате целенаправленных действий при решении поставленной задачи.

Информационная деятельность связана с созданием информационных моделей всех объектов и явлений природы и общества, участвующих в человеческой деятельности, а также с созданием моделей самой этой деятельности.

Как известно, информация как продукт интеллектуальной деятельности человека является ресурсом и с течением времени накапливается, хотя возможны и потери информационных ресурсов.

В процессе познавательной деятельности мы, так или иначе, сталкиваемся с процессом использования накопленного знания, которое становится ценным лишь тогда, когда становится доступным широкому

кругу пользователей. В настоящее время, объем информационных потоков, несущих эти знания, существенно увеличился, поэтому стала актуальной задача информатизации различных видов человеческой деятельности. Данное направление включает в себя развитие аппаратных средств и информационных технологий [1-3].

**Информационные технологии (ИТ)** – совокупность методов и средств реализации информационных процессов в различных областях человеческой деятельности. Иначе говоря, ИТ есть способ реализации информационной деятельности.

К современным ИТ относят:

- развитие глобальных информационных систем;
- внедрение систем автоматизированной обработки информации;
- развитие систем и средств дистанционного доступа;
- интегрирование гетерогенных систем;
- развитие систем искусственного интеллекта и т.д.

Долгое время преобразование информации и принятие решений являлось функцией человека. Сейчас, когда рост объемов информационных потоков привел к тому, что они превысили объемы усвояемости и обрабатываемости информации человеком, возникла проблема повышения эффективности процессов преобразования информации, определяемая следующими причинами:

- любая информация ценна только в процессе ее использования и при резком возрастании объемов информации принятие решений становится затрудненным, а также возрастает время обработки информационного массива;

- усложнение внутренней структуры системы, появление суперсистем, включающих целые совокупности систем, интеграция гетерогенных систем также приводит к резкому увеличению объемов

информационных потоков и времени на их обработку;

- расширение сфер применения ИТ приводит к возникновению новых систем, что, в свою очередь, является дополнительным источником увеличения информационных потоков;

- повышение сложности задач, требуемой для их решения точности и оперативности, приводит к опережающему росту сложности управления по отношению к росту возможностей обработки информации и так далее.

Определим два основных пути развития ИТ, обеспечивающих повышение эффективности процессов преобразования информации в информационных и информационно–управляющих системах:

- совершенствование технических средств автоматизации на основе применения высокопроизводительных вычислительных устройств и систем, что приводит к повышению скорости обработки информации вне зависимости от характера преобразуемой информации;

- совершенствование и расширенное внедрение программного обеспечения.

Для реализации указанных путей необходимо наличие наиболее общих подходов к решению стоящих задач, инвариантных к конкретной содержательной стороне задачи и техническим средствам ее реализации[1-3].

Для информационных систем эта задача обостряется в связи с развитием научного знания, существенным увеличением его объемов, когда уже в рамках узких, под отраслевых вопросов объемы процессов восприятия нового знания превышают возможности человека, не говоря уже о возможностях использования межотраслевого опыта. При этом является рациональным решение, когда сочетаются наиболее общие подходы к решению проблемы с их конкретной технической реализацией. Возможность рассматривать любую систему, абстрагируясь от ее технической реализации, возможность переноса опыта по разработке и исследованию систем, решающих один круг задач, к системам, предназначенным решения задач в

иной области, говорит об открытости, как самих систем, так и о принципах и подходах к их построению и исследованию, которые будут сформулированы ниже.

## 1.2. Требования, предъявляемые к информационным системам

Сами по себе средства вычислительной техники не могут осуществить преобразование информации, для этого необходимо наличие прикладного информационного и программного обеспечения, реализующего функции информационной или информационно–управляющей системы (ИС). ИС представляет собой совокупность элементов, находящихся в отношениях и связях между собой и образующих определенную целостность, единство, и предназначенных для осуществления целенаправленного процесса преобразования информации.

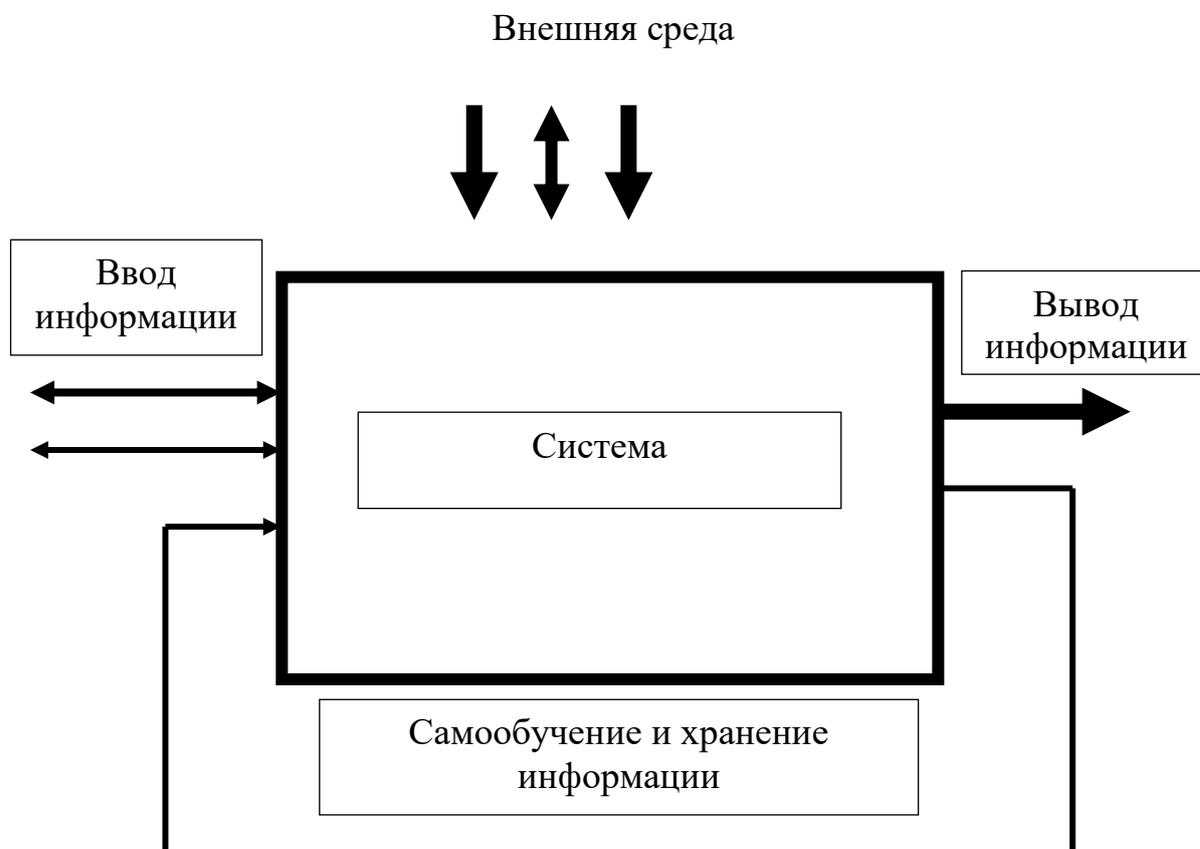


Рис 1.1. Основные функции системы

К основным функциям ИС можно отнести:

- организация интерфейса обмена между технической и информационной системами, а также между ИС и внешней средой;
- организация работы и распределение ресурсов собственно ИС;
- самообучение системы, адаптация к изменяющимся условиям.

ИС должны удовлетворять следующим требованиям:

- обеспечивать достоверность соответствия описаний объектов в ИС по отношению к их реальному состоянию;
- иметь дружественный интерфейс процесса управления,
- обладать возможностями развития и самообучения системы;
- обеспечивать полноту представления информации в системе и во взаимодействии системы с внешней средой, своевременность и обоснованность в выработке соответствующих решений, мобильность ИС при работе в условиях гетерогенных технических средств, реализующих систему, защиту информации в системе;
- обеспечивать реализуемость заданного алгоритма;
- надежность работы в реальных условиях.

Развитие средств компьютерной техники, расширенное их внедрение во все сферы науки, техники, сферы обслуживания и быта привели к необходимости объединения конкретных вычислительных устройств и реализованных на их основе ИС в единые информационно–вычислительные системы (ИВС) и среды[1-3]. При этом возникли следующие проблемы:

- разнородность технических средств ВТ с точки зрения организации вычислительного процесса, архитектуры, системы команд, разрядности процессора и шины данных, ресурсных возможностей, частот синхронизации и так далее, потребовала создания физических интерфейсов, обеспечивающих их совместимость;

- разнородность программных сред, реализуемых в конкретных вычислительных устройствах и системах с точки зрения многообразия операционных систем, различия в разрядности, объемах адресуемой памяти, применяемых языках программирования и так далее, привела к созданию программных интерфейсов между устройствами и системами;

- разнородность реализации одной вычислительной структуры, изготовленной различными производителями, также требовала применения специальных ограничений, либо разработки дополнительных программных и (или) технических средств для интеграции;

- разнородность интерфейсов общения в системе "человек–машина" требовала постоянного переобучения кадров.

Таким образом, необходимость предусмотреть уже на стадии разработки возможность интегрирования разрабатываемого устройства в гомогенные и, особенно, в гетерогенные информационно–вычислительные среды стала актуальной для разработчиков как аппаратных, так и программных средств.

С этой целью, при разработке ИС необходимо соблюдать требование системности, включающее в себя:

1. Систематизацию информационной базы, то есть исключение противоречий и дублирования между отдельными ее частями, обеспечение полного представления информации, согласование времени поиска информации в соответствии со структурой.

2. Организацию и упорядочивание внешних связей ИУС и технических средств автоматизации.

3. Учет условий хранения информации в ИУС.

4. Стандартизацию форм представления информации, форм представления документов, структуры информационной базы, структуры и свойств алгоритмов

Весь процесс разработки можно условно разделить на этапы: анализ системы и разработка ее информационной модели, разработка математической модели (алгоритма), разработка программной модели, разработка документации на ИС.

Рисунок 1.2 отображает типичную структуру технологического процесса ИС или представление ИС как совокупности функциональных подсистем – сбора, ввода, хранения, поиска, распространения информации.

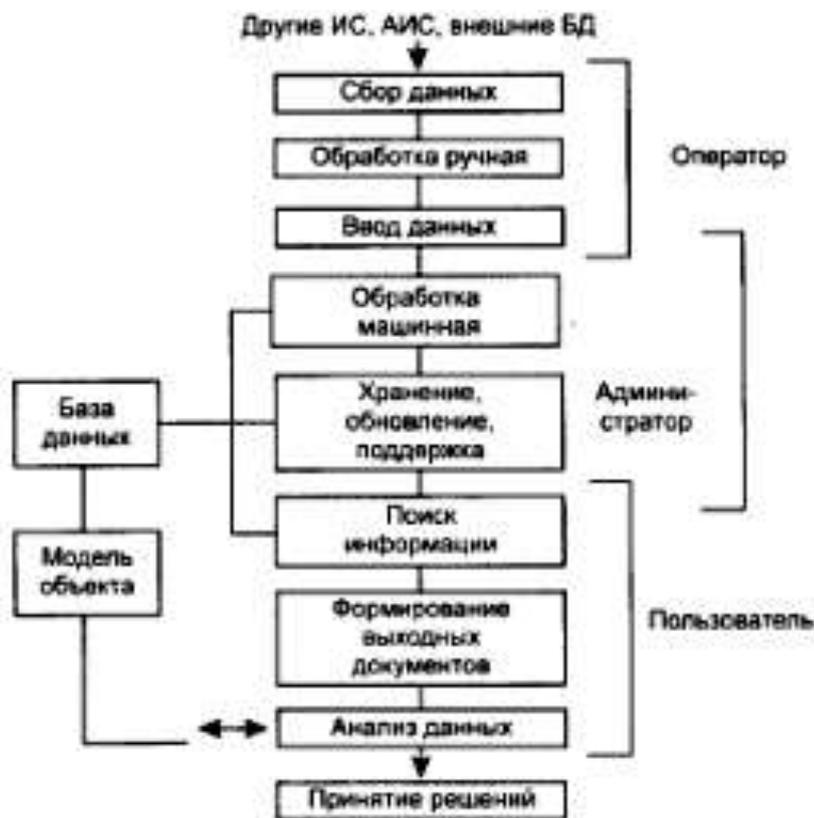


Рис. 1.2. Основные технологические процессы ИС

Некоторые компоненты данной структуры являются необязательными:

1. Модель объекта может отсутствовать либо отождествляться с базой данных, которая часто интерпретируется как информационная модель предметной области, структурная (для фактографических и табличных) или содержательная (для документальных). В экспертных системах в качестве модели объекта (предметной области) фигурирует база знаний, представляющая собой процедурное развитие понятия БД

2. Модель объекта и БД могут отсутствовать (а соответственно и процессы хранения и поиска данных), если система осуществляет динамическое преобразование информации и формирование выходных

документов без сохранения исходной, промежуточной, результирующей информации. Если также отсутствует преобразование информации, то подобный объект не является ИС.

3. Процессы ввода и сбора данных являются необязательными, поскольку вся необходимая и достаточная для функционирования ИС информация может уже находиться в БД и составе модели [1-3].

В более общем случае, учитывая специфику организации, управления и технологии выполнения каждой из указанных функций в ИС целесообразно выделять три самостоятельных функциональных подсистемы:

**Подсистема отбора информации.** Информационная система может обрабатывать/перерабатывать только ту информацию, которая в нее введена. Качество работы ИС определяется не только ее способностью находить и перерабатывать нужную информацию в собственном массиве и выдавать ее пользователю, но и способностью отбирать релевантную информацию из внешней среды. Такой отбор осуществляет подсистема отбора информации, которая накапливает данные об информационных потребностях пользователей ИС (внутренних и внешних), анализирует и упорядочивает эти данные, образуя информационный профиль ИС. Аналогично на основании данных о потоках информационной среды формируется описание входных потоков информации.

При заданном критерии качества функционирования ИС и соответствующей системы ограничений в процессе управления ИС решается задача оптимизации комплектования информационного массива ИС, которая определяет алгоритм (или оператор) отбора информации. Указанный оператор осуществляет преобразование входных потоков в информационный массив ИС. К сожалению, многие действующие ИС слабо придерживаются описанной процедуры отбора документов. Отбор информации, как правило, носит слабоуправляемый характер, базируется на интуиции специалистов.

Это является следствием сложности и слабой структурируемости как собственно процессов отбора, так и управления этими процессами.

Функции именно этой подсистемы ИС практически не поддаются автоматизации. Исключение составляют только ИС информационного обеспечения управления технологическими процессами и техническими системами.

**Подсистема ввода, обработки/переработки и хранения информации** осуществляет преобразования входной информации и запросов, организацию их хранения и переработки с целью удовлетворения информационных потребностей абонентов ИС.

Реализация функций данной подсистемы предполагает наличие аппарата описания информации (ИПЯ, систем кодирования, ЯОД и т. д.), организации и ведения информации (логическая и физическая организация, процедуры ведения и защиты информации и т. д.), аппарата обработки и переработки информации (алгоритмы, модели и т. д.).

Все три указанные составляющие определяются двумя параметрами ИС: характером обрабатываемой информации и функциями ИС.

Документальные ИС для описания информации используют ИПЯ и систему индексирования, методология построения и использования которых существенно отличается от методологии и принципов использования ЯОД, обеспечивающих описание данных в фактографических ИС. Логическая организация данных фактографических ИС имеет мало общего с организацией информации в документальных ИС. Наконец, различны и аппараты обработки и переработки документальной и фактографической информации. Если в фактографических ИС преимущественно используются математические алгоритмы, то в документальных – эвристические процедуры, требующие затрат интеллектуальной энергии [1-3].

**Подсистема подготовки и выдачи информации** непосредственно реализует удовлетворение информационных потребностей пользователей ИС

(внутренних и внешних). Для выполнения этой задачи подсистема проводит изучение и анализ информационных потребностей, определяет формы и методы их удовлетворения, оптимальный состав и структуру выходных информационных продуктов, организует сам процесс информационного обеспечения и сопровождения. Выполнение указанных функций требует наличия аппарата описания и анализа информационных потребностей и их выражения на языке ИС (в том числе ЯОД, ИПЯ, языке индексирования и т. д.), а также аппарата непосредственно информационного обеспечения (процедуры поиска и выдачи информации, языки манипулирования данными и т. д.).

Все эти и многие другие составляющие рассматриваемой подсистемы, выполняя одинаковые функции в ИС разных типов, тем не менее существенно отличаются между собой. Особенно заметно это различие при сравнении документальных и фактографических ИС.

Из предыдущего рассмотрения следует, что многие функции различных подсистем ИС дублируются или пересекаются, что является предметом оптимизации при проектировании ИС. Автоматизация ИС в связи с этим сопровождается перераспределением элементов ИС.

Автоматизация предполагает формализованное представление (структуризацию) как функций ИС, так и самой обрабатываемой в ИС информации, которое и позволяет осуществлять ввод, обработку/переработку, хранение и поиск информации с использованием ЭВМ.

Однако любая формализация характеризуется тем или иным уровнем адекватности создаваемого образа реальной действительности (модели) самой действительности. Причем, адекватность модели реальной действительности определяется как свойствами самой действительности, так и возможностями используемого аппарата ее формализованного представления.

С этой точки зрения "уровень автоматизации" ИС тесно связан со "степенью структурируемости" как самой информации, являющейся предметом обработки, хранения и т. д., в ИС, так и самих функций ИС (обработки, хранения и т. д.).

В соответствии с уровнем современных знаний в области формализованного представления информации можно различать информацию трех уровней структурируемости:

1. Жесткоструктурируемая информация – информация, формализованное представление которой современными средствами ее структурирования (в частности, языками описания данных) не приводит к потере адекватности создаваемого образа информации (модели) самой исходной информации. Жесткоструктурируемую информацию будем в дальнейшем называть данными.

2. Слабоструктурируемая информация – информация, формализованное представление которой современными средствами описания информации (в частности, ИПЯ) приводит к значительным потерям адекватности модели информации самой исходной информации. Обработка и поиск такой информации предполагает специальные меры по оценке степени неадекватности модели информации. (В АИПС этой цели служат меры смысловыразительной способности (семантической силы) ИПЯ).

3. Неструктурируемая информация – информация, для которой в настоящее время не существует средств ее формализованного представления с приемлемым на практике уровнем адекватности. Средства представления такой информации должны обладать высокими смысловыразительными способностями. Разработка таких средств в настоящее время идет по линии создания языков описания знаний и ИПЯ с высокой семантической силой.

Приведенная классификация информации по степени ее структурируемости достаточно условна. Однако сама идея учета

структурируемости информации оказывается полезной при анализе сущности современных АИС.

Если с этих позиций рассмотреть функции подсистем ИС, то нетрудно видеть, что большинство жестко структурируемых функций сосредоточено в подсистеме ввода, обработки/переработки и хранения информации. Две другие подсистемы связаны с реализацией, в основном, слабо структурируемых и не структурируемых функций [1-3].

Легкость автоматизации функций второй подсистемы ИС на основе использования электронно–вычислительной и телекоммуникационной техники ввода, обработки, хранения и передачи информации привела к неоправданно быстрому и вседвоящему развитию этих составляющих ИС в ущерб развитию двух других (не менее, а может быть и более важных) ее составляющих. В большинстве современных АИС эти две подсистемы настолько неразвиты, что по сути дела это уже не АИС, а организационно–обособленные подсистемы ввода, обработки, хранения и поиска информации. В дальнейшем будет показано, что говоря об этих системах, целесообразно называть их не АИС, а банки данных или АИПС.

## **ГЛАВА 2. ПРОГРАММНЫЕ СРЕДСТВА ДЛЯ СОЗДАНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ С ИНТЕРАКТИВНЫМИ УСЛУГАМИ**

### **2.1. Обзор инструментальных средств**

В настоящее время существует большое количество готовых инструментальных средств, применяемых для создания компьютерных средств обучения. Они предоставляют среду для обработки и редактирования элементов продуктов мультимедиа, включая графические изображения, звуковые элементы, анимацию и видеоклипы; мультимедиа педагогического программного средства в целом, включая структурное представление сценария и его реализацию. Существующие инструментальные средства и технологии для разработки компьютерных средств обучения позволяют работать с ними не только опытным программистам, но и тем, кто никогда ранее не сталкивался с написанием программ.

Инструментальные средства - комплекс компьютерных программ, предоставляющих пользователям, не владеющим языками программирования, создавать свои компьютерные средства интерактивные услуги через сайта.

Инструментальные средства предоставляют педагогу следующие возможности работы:

- готовить разностороннюю информацию (теоретический и демонстрационный материал, практические задания, вопросы для тестового контроля);
- формировать сценарий создания определенного интерактивной услуги (ИУ);
- значительно сокращать время на подготовку ИУ и проведения занятий (группового контроля);
- реализовать через созданные ИУ свою методику изложения материала и проведения общения.

Многие авторы выделяют следующие группы инструментальных систем, которые можно использовать для создания ИУ.

Рассмотрим возможный вариант классификации инструментальных программных систем разработки ИУ:

1) специализированные программные средства, которые предназначены для быстрой подготовки определенных типов гиперссылочных или мультимедийных приложений (презентаций, анимационных роликов, публикаций в сети Интернет, звуковых записей и др.).

К специализированным программным средствам обычно относят:

- Macromedia Flash;
- Macromedia Dreamweaver;
- Microsoft FrontPage;
- Microsoft PowerPoint;
- EasyHelp;

2) авторские средства разработки предназначены для создания программных средств учебного назначения.

К авторским средствам разработки чаще всего относят:

- Adobe Authorware;
- HyperMethod;
- ToolBook Assistant;
- Web Course Builder;
- другие;

3) универсальные языки программирования позволяют создавать КСО, но для разработки последних необходимо владеть достаточно хорошими навыками программирования.

К универсальным языкам программирования, используемым для разработки КСО, относят:

- Delphi;
- Javascript;

- Perl;
- Си ++;
- другие.

В настоящее время разработаны не только готовые инструментальные системы (заготовки, шаблоны) для создания КСО, но имеется достаточное количество готовых инструментальных средств, позволяющих создавать современные, достаточно гибкие средства обучения, контроля, моделирующие и демонстрационные программы, сайты, электронные гиперссылочные учебники и многое другое. Представим некоторые из них.

Среда Macromedia Authorware - это лучшая на сегодняшний день визуальная среда разработки интерактивных мультимедийных обучающих приложений. Инструментальная среда позволяет создавать очень интересные по организации сетевые мультимедийные интерактивные учебные пособия.

Возможности Authorware:

- визуальное создание сценария проекта (возможно изменение местоположения пунктов простым перетаскиванием мышкой);
- выбор типа проекта (обучающий, тестирующий, демонстрационный);
- включение в основной проект внешних приложений;
- подготовка анимации;
- создание контролирующих приложений;
- другое.

Macromedia Dreamweaver - программная среда для создания электронных документов и Web-страниц. Разработчику Web-проекта предлагается гибкий и мощный инструментарий управления сайтом, включая встроенный полноценный инструмент FTP- клиент, визуальные карты сайтов и контроль над ссылками. Dreamweaver - инструмент для создания web-сайтов, электронных средств обучения помогает в построении ярких, интерактивных и эффективных интерфейсов. Dreamweaver позволяет

создавать объекты и страницы с использованием языка DHTML, поддерживает каскадные таблицы стилей, а также слои и действия JavaScript.

Возможности Dreamweaver значительно облегчают процесс создания страниц и управления сайтом. Macromedia Dreamweaver объединяет признанные средства визуальной разметки электронного документа с функциями быстрой разработки web-приложений. Dreamweaver UltraDev с возможностями редактора Macromedia HomeSite легко интегрируется со средой Flash MX.

Инструментальное средство Macromedia Flash. Macromedia Flash - это профессиональный программный продукт, первоначально ориентированный на создание интерактивной анимации для системы World Wide Web.

Выделяют обычно следующие возможности и особенности Macromedia Flash:

- универсальность;
- возможность интеграции в другие приложения;
- наличие развитых мультимедийных возможностей: средства создания анимаций, демонстраций, лабораторных работ;
- возможность использования слоев, для разбиения перекрывающихся объектов;
- возможность создания мультфильмов, обучающих игр, электронных симуляций (для обучения работе в других приложениях);
- возможность вставки ранее созданных проектов в текущий проект;
- возможность работы созданного программного средства под любой операционной системой.

Средство Microsoft FrontPage входит составной частью в полный пакет Microsoft Office - программный продукт широкого использования для создания Internet документов, использует технологию программирования клиент- сервер, включает в себя клиентскую и серверную части и предоставляет возможность разработки электронных образовательных

ресурсов, сайтов в целом и установки его на большинство популярных серверов.

Microsoft Office FrontPage представляет собой мощный редактор Web-документов, позволяющий размещать на Web-страницах текстовую и графическую информацию, а также эффекты мультимедиа (звук, видео, анимацию). В системе Microsoft FrontPage удобно и достаточно просто создавать электронные учебные пособия. Для использования Microsoft FrontPage не требуется знания языка разметки электронных документов HTML. В ходе создания и редактирования документов программа автоматически формирует и добавляет символы (теги) языка HTML в кодовое описание страницы (при вводе и форматировании текста, добавлении рисунков, таблиц, ссылок и других элементов страниц). С помощью Frontpage может сконструировать вполне работоспособный web-узел или ЭУП даже новичок, совершенно незнакомый с языком разметки электронных документов HTML. В руках опытного пользователя Frontpage становится мощным инструментом, позволяющим строить web-узлы любого уровня сложности. Одна из особенностей средства Frontpage заключается в его умении представлять создаваемые страницы в режиме WYSIWYG (what you see is what you get — что видите, то и получаете). Это означает, что в браузере созданные страницы будут выглядеть так же, как и при создании в среде Frontpage. Следует отметить, что Microsoft FrontPage предоставляет профессиональные средства разработки и проектирования, подготовки и публикации данных, которые требуются для создания динамических Web-узлов различного назначения.

Среда Microsoft PowerPoint является наиболее простым средством разработки обучающих и информационных материалов (лекций, докладов, презентаций). По количеству изобразительных и анимационных эффектов эта среда становится вровень со многими авторскими инструментальными средствами мультимедиа. Одним из серьезных недостатков, созданных в этой

среде КСО, является открытость созданного средства обучения, его незащищенность от несанкционированного доступа.

Среда ToolBook Assistant поддерживает огромное количество разнообразных медиа-форматов, включая звук, анимацию, цифровое видео, сжатые изображения. ToolBook Assistant - это набор специализированных авторских средств для создания мультимедиа приложений обучающего характера. В его состав входят ToolBook Instructor, ToolBook Actions Editor и ToolBook Simulation Editor, при помощи которых можно быстро и эффективно создать интерактивное содержание с набором мультимедийных объектов любых форматов.

Кроме указанных инструментальных сред подготовки компьютерных средств обучения есть большое количество других программных продуктов, соответствующих целям создания приложений обучающего характера. Более развернутая информация по выбору инструментальных средств для разработки КСО представлена в работе. Достоинства и недостатки инструментальных средств создания КСО представлены в таблице 11 на основе рассмотренных В.В. Запорожко в ранее указанной работе. В Интернете представлено достаточно большое количество сайтов и порталов, содержащих информацию об имеющихся инструментальных средствах, позволяющих создавать компьютерные средства обучения и управления деятельностью образовательных учреждений и для решения других задач. Необходимое инструментальное средство можно найти в электронном каталоге разных выпусков <http://www.catalog.iot.ru> и других изданиях.

## **2.2. Разработка интерактивных сайтов**

Интерактивный сайт — большинство современных сайтов относится именно к этой разновидности. Что представляет собой интерактивный сайт? Каковы преимущества интерактивных сайтов? Как сделать интерактивный сайт?

Услышав словосочетание «интерактивный сайт», многие ассоциируют его с различными flash-эффектами, активируемыми в зависимости от местоположения курсора мыши посетителей сайта. Однако этот стереотип несколько неверен.

На самом деле, под интерактивным следует понимать сайт, контент которого формируется «на лету», на основе данных, полученных от посетителя сайта и обработанных сервером. Поэтому понятие «интерактивный» можно отнести ко всем сайтам, имеющим формы отправки сообщений, онлайн-анкеты и опросы, регистрационные формы, счетчики посещений, формы для онлайн-заказов и другие подобные элементы.

Для реализации интерактивных «свойств» сайтов применяются специальные программные коды —серверные скрипты. Именно серверные скрипты производят обработку полученных от посетителей сайта данных и формируют ответную html-страницу.

Для написания серверных скриптов применяются серверные языки веб-программирования, такие как PHP, Perl, ASP.NET. Выполняется серверный скрипт на стороне сервера: посетитель не видит исходного программного кода выполняемого скрипта, а получает только готовый ответ.

Создать интерактивный сайт - значит создать сайт, умеющий «общаться» со своими посетителями. Простой статический сайт доступен лишь для просмотра. Для того, чтобы связаться с администрацией сайта или же сделать заказ, посетителю необходимо позвонить по указанному в контактных данных телефону, написать письмо на e-mail или же отправить факс.

В то же время, интерактивный сайт позволяет поддерживать связь между администрацией и посетителями в режиме реального времени без каких-либо вспомогательных средств — для этого есть системы онлайн-консультаций, онлайн-чатов (системы мгновенного обмена сообщениями прямо на сайте) и другие подобные инструменты.

Различные интерактивные компоненты могут быть эффективно использованы при создании сайтов для бизнеса. Интерактивный сайт для бизнеса позволяет владельцам сайта получить дополнительный доход, благодаря активному привлечению клиентов, а посетителям — удобный и доступный сервис, который без сомнений, выгодно отличит компанию на фоне ее конкурентов.

Интернет-магазин — наиболее типичный пример интерактивного сайта для бизнеса. В интернет-магазине посетитель может сам сформировать свой заказ, выбрав необходимые товары и указав количество приобретаемых единиц.

Так называемая «корзина» позволяет рассчитать суммарную стоимость заказа, сформировать непосредственно счет для оплаты заказа различными средствами. Все это выполняется в режиме реального времени и пользователь сам может изменять наименования, характеристики (цвет, количество) товаров, которые он приобретает.

Более того, большинство современных интернет-магазинов позволяют своим покупателям отслеживать текущее состояние заказа: принят в обработку, доставлен на склад, доставлен покупателю и т. п.

Как сделать интерактивный сайт?

Создание интерактивных сайтов может включать следующие этапы:

- определение целей и задач интерактивного сайта;
- разработка концепции интерактивного сайта;
- выбор средств реализации интерактивных возможностей сайта;
- разработка дизайн-макета сайта, верстка шаблона, программная реализация интерактивных компонентов, наполнение сайта контентом;
- юзабилити-тестирование сайта, проверка интерактивных компонентов на соответствие поставленным целям и задачам;
- размещение сайта в web и дальнейшая поддержка и развитие сайта.

Среди интерактивных услуг Интернета, которые могут быть полезны в дистанционном учебном процессе, следует особо выделить:

- электронную почту, посредством которой учащиеся могут общаться с учителями, со своими партнерами;
- список рассылки, который используется преподавателем или кем-то из учащихся, когда возникает необходимость дать информацию сразу всем участникам курса;
- телеконференцию в режиме реального времени (IRC — Internet Relay Chat) или чат;
- NetMeeting – программу, входящую в комплекс Microsoft Internet Explorer. Эта программа позволяет осуществлять интерактивный обмен мнениями членов группы (малой группы или общей группы) с демонстрацией обсуждаемых материалов (в том числе графику). Результаты обсуждения можно сохранять в виде отдельных файлов для последующего анализа;
- телеконференции с отсроченным доступом (off-line).

Чаще всего до сих пор используется электронная почта, которая позволяет пересылать файлы любого объема с использованием любой иллюстрации: визуальной, звуковой, мультимедийной. Приходится принимать во внимание время перекачки информации даже в архивированном виде.

### **2.3.Проектирование автоматизированных информационных систем**

Весьма острая проблема, как отмечают зарубежные специалисты, связана с необходимостью оптимизации стоимости образовательных услуг, предлагаемых через Интернет. Баланс между первоначальными затратами, количеством студентов и качеством используемого методического обеспечения может решить проблему экономической эффективности дистанционного обучения и предоставление услуги. Авторы аналитического

обзора со ссылкой на Р. Оустона отмечают противоречие, которое заключается в том, что, с одной стороны, основные виды программного обеспечения для web-браузеров и серверов поставляются практически бесплатно или за минимальную плату, основные расходы поэтому падают на приобретение аппаратных средств и на обеспечение канала связи. С другой стороны, если в дистанционной форме обучается небольшое количество учащихся, то окупаемость затраченных средств растягивается на длительный период.

Модель жизненного цикла (ЖЦ) - структура, содержащая процессы, действия и задачи, которые осуществляются в ходе разработки, функционирования и сопровождения программного продукта в течение всей жизни системы, от определения требований до завершения ее использования. Существует несколько моделей и стандартов, в той или иной степени регламентирующих жизненный цикл, большинство из них относятся к заказному программному обеспечению и кроме непосредственно ЖЦ регламентируют также и процессы разработки:

Решить проблему повышения эффективности управления производством в современных условиях невозможно без внедрения новейших информационных технологий и современных методов управления. Наиболее перспективным направлением сегодня является разработка тиражируемых отраслевых систем управления. Рассмотрим методику проектирования автоматизированных информационных систем управления предприятием, которая состоит, по нашему мнению, из следующих этапов.

- Обследование объекта автоматизации (анализ) и формулирование требований пользователей к системе управления.

- Постановка целей. Анализ существующих методов и средств автоматизации аналогичных объектов и формулирование на основании требований пользователя достижимых целей функционирования системы управления. Цели должны быть четкими, явными и измеримыми. Цели

должны определять: общее назначение системы, определение разных групп пользователей и их роли, подробное перечисление функций системы, виды необходимой документации, параметры эффективности (производительности), совместимость с другими продуктами и стандартами, конфигурации аппаратуры, средства обеспечения безопасности, методы и средства настройки и обслуживания, методы обеспечения надежности системы. Цели не должны конфликтовать между собой, так как ими необходимо руководствоваться для выработки компромиссных решений на следующих этапах проектирования.

- Разработка архитектуры системы (декомпозиция функциональной структуры и определение связей между ее элементами). Выделение уровней управления, подсистем, комплексов задач, задач и функций управления.

- Разработка инфологической модели системы, описывающей статику и динамику объекта. Формализация моделей состояния объекта, материальных, финансовых и информационных (управляющих) потоков и их взаимодействия между собой.

- Разработка системы классификации объектов учета и управления и идентификации их параметров. Словари описывают основные понятия предметной области системы, необходимые для разработки стандартных алгоритмов обработки данных. Классификаторы описывают структуру объекта (подразделения, сотрудники, должности), внешней среды (клиенты, районы, пункты погрузки/разгрузки), характеристики материальных потоков (партии, фонды, ед. измерения, показатели качества, типы цен, виды оплаты). Типовые операции описывают алгоритмы управления (обработки информации).

- Разработка информационной модели системы (проектирование структур баз данных и их связей).

- Синтез структуры программного обеспечения (агрегирование системы). При объединении отдельных функций управления в программные

модули необходимо стремиться к высокой "прочности" и слабому "сцеплению" модулей. Прочность и сцепление модуля являются, соответственно, мерами его внутренних и внешних связей. В зависимости от назначения модулей необходимо стремиться либо к их функциональной прочности (объединение взаимосвязанных функций управления), либо к информационной прочности (объединение функций, выполняемых на ограниченном подмножестве информационного пространства системы).

- Выбор метода сборки и тестирования системы. Известно несколько методов сборки и тестирования сложных программных систем: восходящий, нисходящий, модифицированный нисходящий, большого скачка, метод сэндвича, модифицированный метод сэндвича. Рекомендуется использовать для тестирования системы модифицированный метод сэндвича, при котором модули нижних уровней управления тестируются снизу вверх, а модули верхних уровней управления сначала тестируются автономно, а затем собираются в агрегаты нисходящим методом. Преимуществами предложенного метода являются: высокий параллелизм в программировании модулей, небольшое количество заглушек, минимальное время появления рабочей версии системы. Отметим, что от выбранного метода сборки и тестирования сильно зависит последовательность проектирования и программирования отдельных модулей. Поэтому метод сборки системы необходимо выбрать до начала этапа проектирования модулей.

- Проектирование модулей. Разработка внешних спецификаций, описывающих сопряжения (связи) между модулями, и проектирование логики (алгоритмов) модулей.

- Программирование модулей на выбранных программных средствах. При программировании необходимо помнить, что текст программы необходим для общения с людьми, а не с машиной. Важность этого утверждения станет очевидна, когда наступит этап сопровождения системы. Для повышения надежности программного обеспечения необходимо

использовать при программировании метод взаимного недоверия модулей, то есть каждый модуль системы должен относиться с определенной долей недоверия, в разумных пределах, к полученным входным данным и проверять их перед обработкой.

- Интеграция (сборка) системы в соответствии с выбранным методом и ее тестирование. Этапы тестирования: автономное тестирование - контроль отдельного программного модуля изолированно от других модулей, тестирование сопряжений - контроль сопряжений между частями системы, тестирование функций - контроль выполнения системой автоматизируемых функций управления, комплексное тестирование - испытание поведения системы по отношению к исходным целям, тестирование приемлемости - проверка соответствия системы требованиям пользователей. Тестирование - процесс выполнения программы с целью найти в ней ошибки. Существует два подхода к проектированию тестов - тестирование по отношению к спецификациям (не заботясь о тексте программы) и тестирование по отношению к тексту программы (не заботясь о спецификациях). Разумный компромисс лежит где-то посередине, смещаясь в ту или другую сторону в зависимости от функций, выполняемых конкретным модулем. Также отметим, что стоимость этапа тестирования составляет до 25% от общей стоимости затрат на разработку системы.

- Разработка методического обеспечения. Руководства пользователей, инструкции по эксплуатации, технологические инструкции.

- Внедрение системы на объекте.

- Сопровождение системы: устранение ошибок и замечаний пользователей, разработка дополнительных режимов и функций управления, функциональное расширение системы. В соответствии со спиральной моделью жизненного цикла программного обеспечения осуществляется переход на 1 - 10 этапы проектирования системы.

Особо отметим, что этап сопровождения является самым дорогим

этапом, его стоимость оценивается экспертами в 50 % от общей стоимости разработки системы. Это можно объяснить тем, что на самом деле этот этап не является самостоятельным, а объединяет группу перечисленных выше этапов проектирования на следующих за этапом внедрения системы витках спирали жизненного цикла программного обеспечения.

**Анализ существующих систем управления базами данных и выбор наилучшей.** Современные СУБД в основном являются приложениями Windows, так как данная среда позволяет более полно использовать возможности персональной ЭВМ, нежели среда DOS. Снижение стоимости высокопроизводительных персональных компьютеров обусловил не только широкий переход к среде Windows, где разработчик программного обеспечения может в меньшей степени заботиться о распределении ресурсов, но также сделал программное обеспечение ПК в целом и СУБД в частности менее критичными к аппаратным ресурсам электронно-вычислительной машины.

Среди наиболее ярких представителей систем управления базами данных можно отметить: Lotus Approach, Microsoft Access, Borland dBase, Borland Paradox, Microsoft Visual FoxPro, Microsoft Visual Basic, а также баз данных Microsoft SQL Server и Oracle, используемые в приложениях, построенных по технологии “клиент-сервер”. Фактически, у любой современной СУБД существует аналог, выпускаемый другой компанией, имеющий аналогичную область применения и возможности, любое приложение способно работать со многими форматами представления данных, осуществлять экспорт и импорт данных благодаря наличию большого числа конвертеров. Общепринятыми, также, являются технологии, позволяющие использовать возможности других приложений, например, текстовых процессоров, пакетов построения графиков и т.п., и встроенные версии языков высокого уровня (чаще – диалекты SQL и/или VBA) и средства визуального программирования интерфейсов разрабатываемых

приложений. Поэтому уже не имеет существенного значения на каком языке и на основе какого пакета написано конкретное приложение, и какой формат данных в нем используется. Более того, стандартом “де-факто” стала “быстрая разработка приложений” или RAD (от английского Rapid Application Development), основанная на широко декларируемом в литературе “открытом подходе”, то есть необходимость и возможность использования различных прикладных программ и технологий для разработки более гибких и мощных систем обработки данных. Поэтому в одном ряду с “классическими” СУБД все чаще упоминаются языки программирования Visual Basic 4.0 и Visual C++, которые позволяют быстро создавать необходимые компоненты приложений, критичные по скорости работы, которые трудно, а иногда невозможно разработать средствами “классических” СУБД. Современный подход к управлению базами данных подразумевает также широкое использование технологии “клиент-сервер”.

Таким образом, на сегодняшний день разработчик не связан рамками какого-либо конкретного пакета, а в зависимости от поставленной задачи может использовать самые разные приложения. Поэтому, более важным представляется общее направление развития СУБД и других средств разработки приложений в настоящее время.

Рассмотрим более подробно программные продукты компании Microsoft, а именно Visual FoxPro 3.0, Paradox, Visual Basic 4.0, Visual C++, Access 7.0.

Наиболее интересной чертой этих пакетов являются их большие возможности интеграции, совместной работы и использования данных, так как данные пакеты являются продуктами одного производителя, а также используют сходные технологии обмена данными.

FoxPro (фирма Fox Software) обладала исключительно высокими скоростными характеристиками и в этом отношении заметно выделялась среди интерпретирующих систем. Сравнительно с dBaseIV ее скорость в

несколько раз выше и не уступает скорости систем-компиляторов. Практически по всем показателям Fox-программы работают значительно быстрее Clipper-программ. (Напоминаем - речь пока о версии для DOS'а.) Набор команд и функций, предлагаемых разработчиками FoxPro, по мощи и гибкости отвечает любым требованиям к представлению и обработке данных. Может быть реализован максимально удобный и эффективный пользовательский интерфейс. В FoxPro поддерживаются разнообразные всплывающие и многоуровневые меню, работа с окнами и мышью, реализованы функции низкоуровневого доступа к файлам, управление цветами, настройками принтера, данные могут быть представлены с виде «электронных таблиц» и много еще приятностей и удобностей. В «довиндовскую» эпоху FoxPro был самой быстрой, самой удобной и самой мощной СУБД для компьютеров стандарта IBM PC.

## ГЛАВА 3. ЭЛЕМЕНТЫ РАЗРАБОТАННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ С ИНТЕРАКТИВНЫМИ УСЛУГАМИ

### 3.1. Основные элементы информационной системы

Главная страница информационной системы имеет вид рис.1

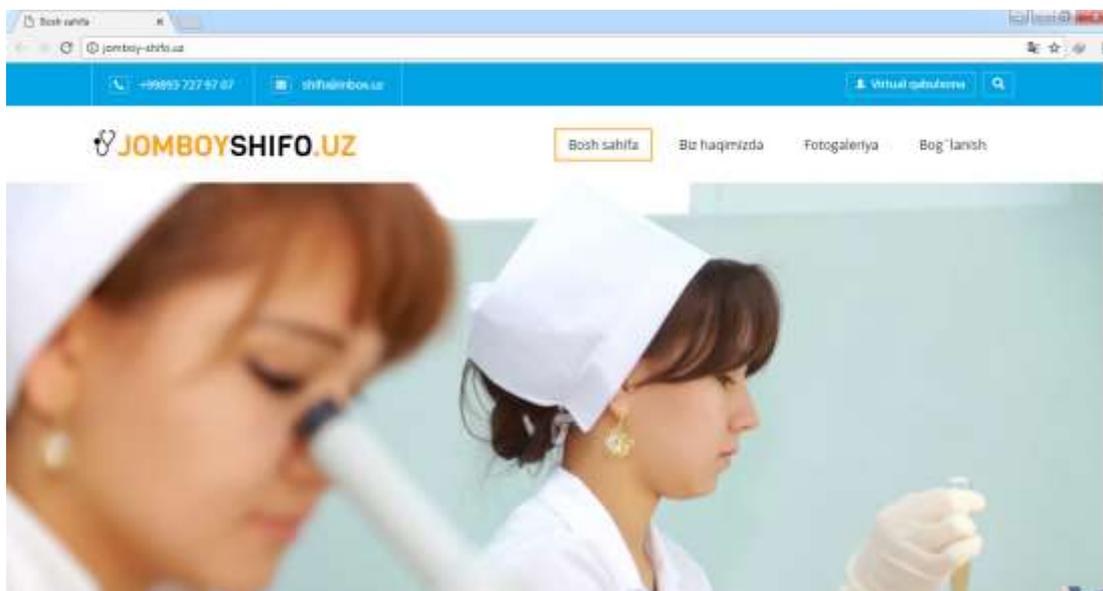


Рис.1. Главная страница информационной системы

На главной странице сайта имеется 6 основных разделений:

- главная страница содержит информацию о больнице;
- Новости – перечень новостей больницы;
- Консультация – содержит информации о первой помощи в домашних условиях
- администрация – сохраняет данные об отделах и их руководителях;
- адрес – содержит адрес поликлиника, также на данной странице можно оставить свои заметки и сообщения.

При разработки системы мы использовали **HTML, CSS, PHP, JQuery, Bootsrap, MySQL, VAZA, Server, Open Server**

На следующем окне показаны основные советы которое пользователи могут соблюдать при заболеваниях. (рис.3.2)



рис.3.2. Страница о больнице

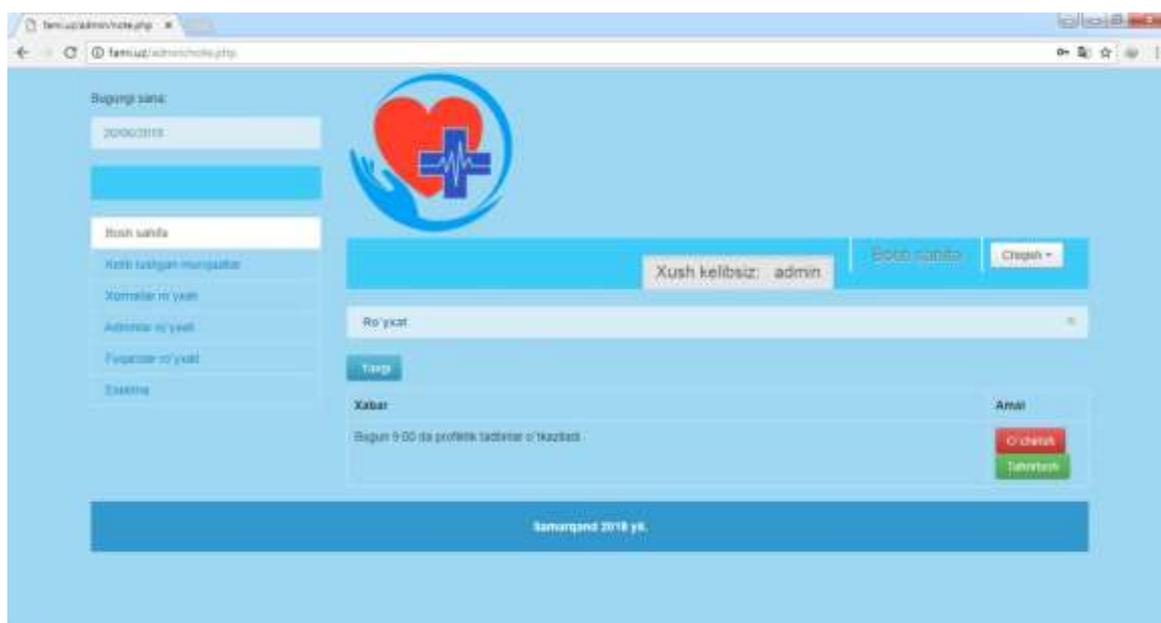


Рис.3. Страница новостей в поликлинике и в области медицины

В следующем окне внесены связь с поликлиникой (рис. 4)

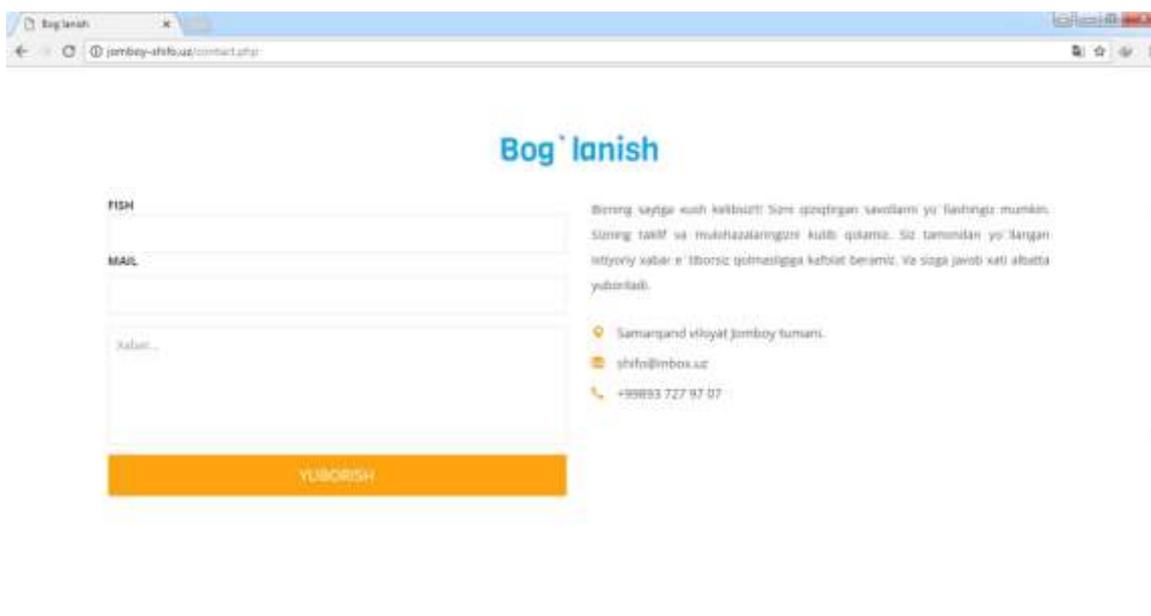


рис.3.4. Страница вызова через электронную почту

Страница галерея содержит фотографии о сегодняшней деятельности больницы.

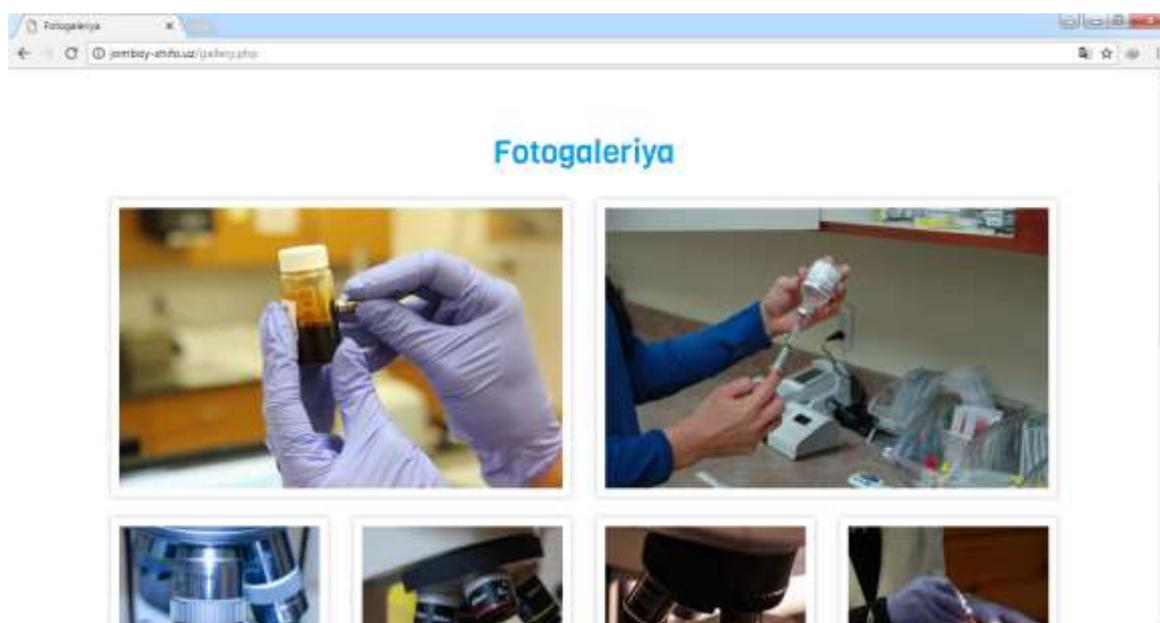


Рис.3.5. Галерея

Следующем окне можно регистрировать пациентов (рис.3.6.)



рис.3.6. Виртуальный прием

Следующим окне показана режим работы специалистов больницы



Рис.3.7. Время приёмов

Данное окно содержит услуги которые имеются в поликлинике (рис.3.9.)

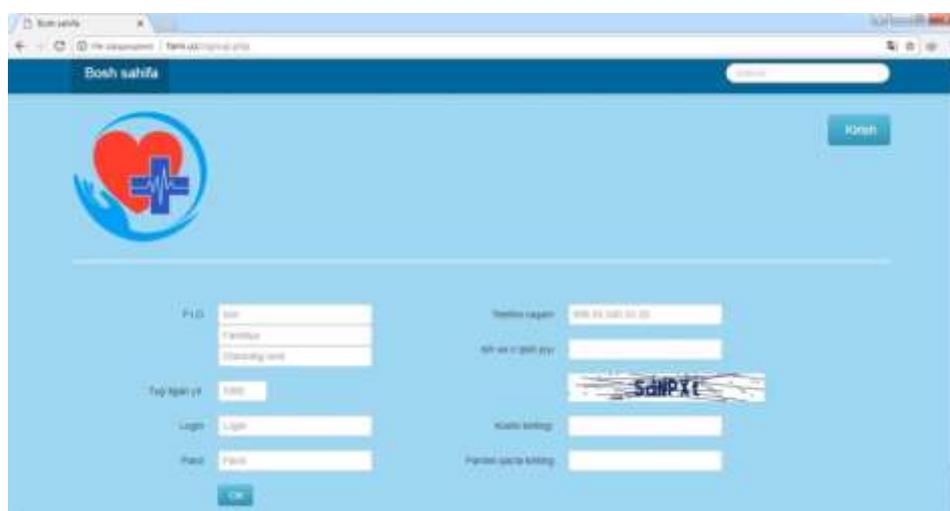


Рис.3.8. Услуги больницы

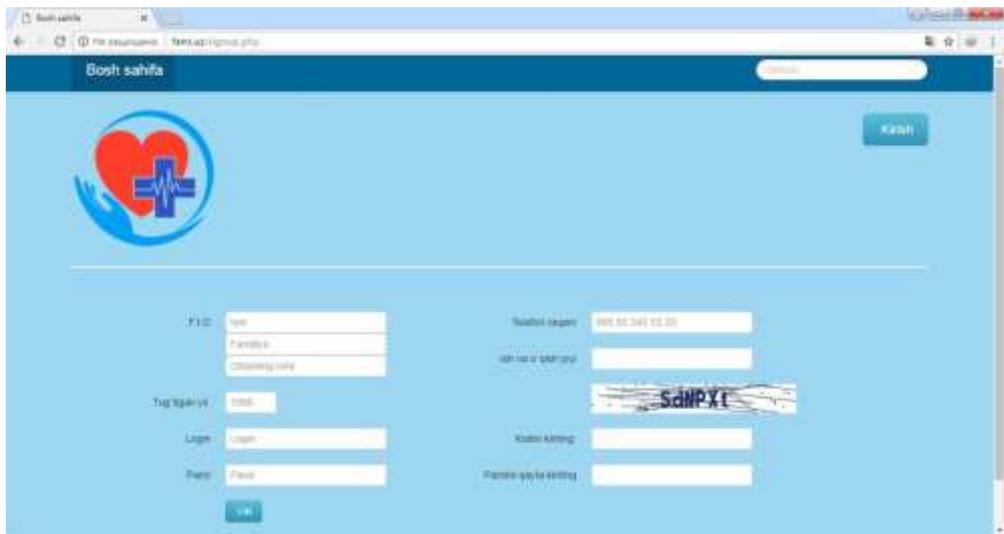


Рис.3.9.Регистрация пациентов больницы



Рис.3.10.Окно авторизации



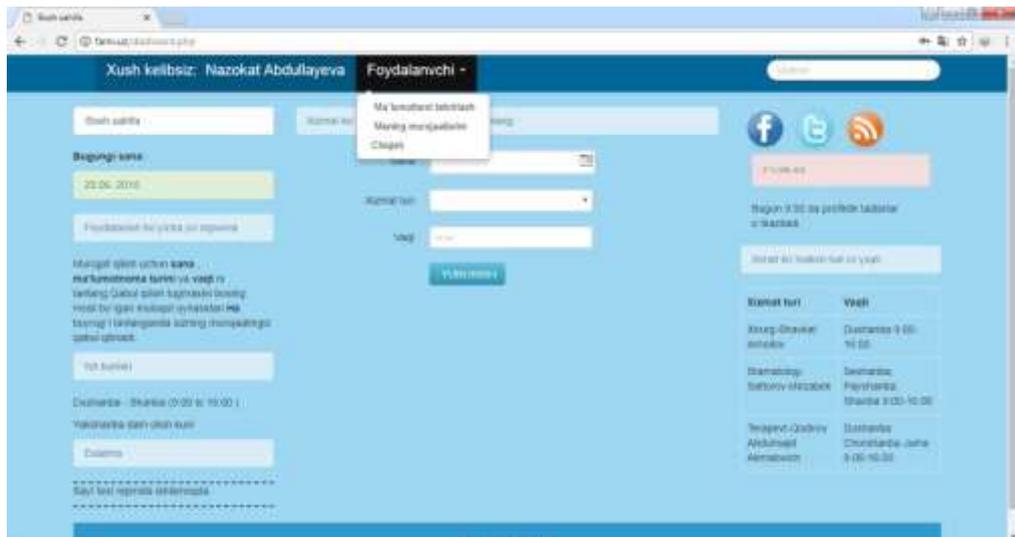


Рис.3.10. Профиль пациента

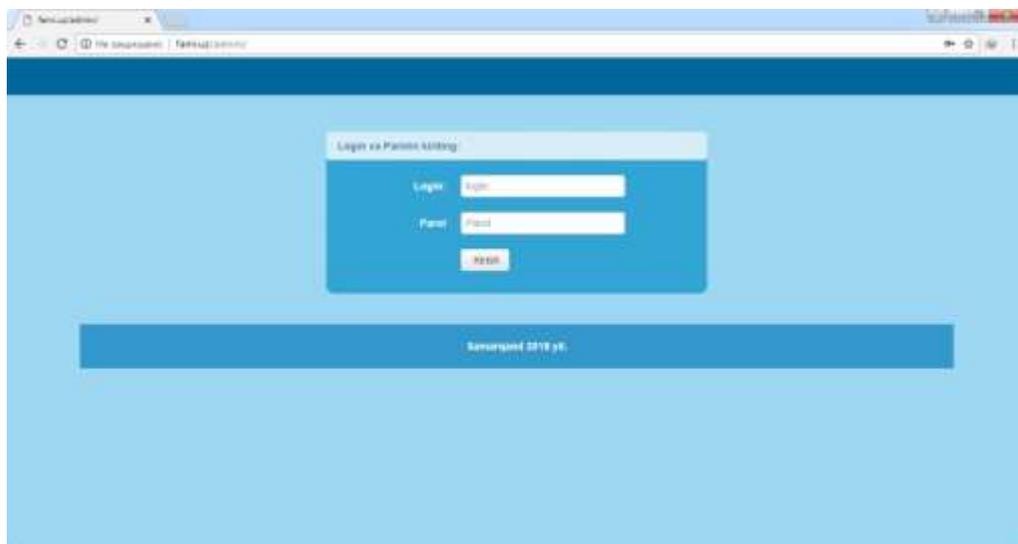


Рис.3.11. Окно авторизации врача информационной системы



Рис.3.12. Профиль врача

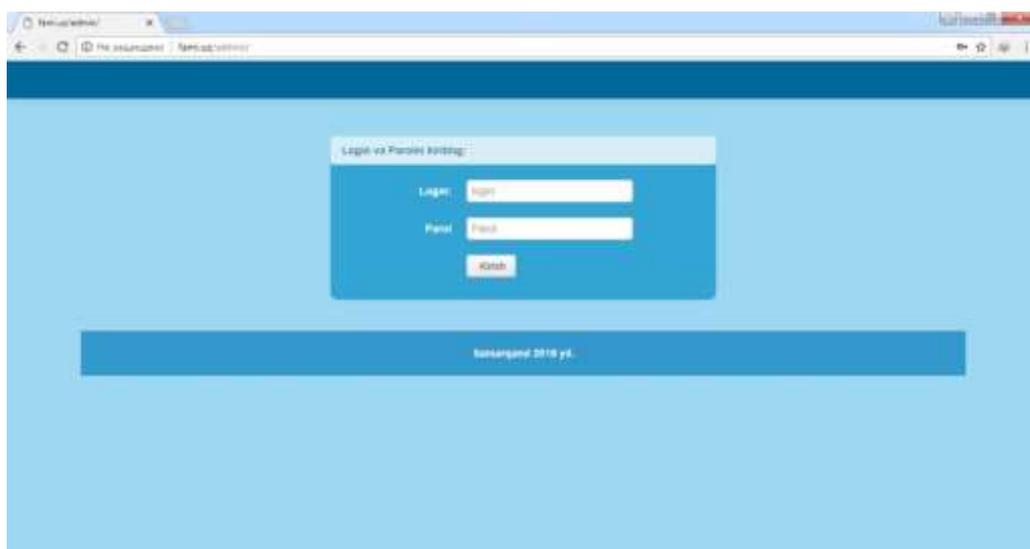


Рис.3.13. Окно авторизации администратора



Рис.3.14. Окно добавление врача



Рис.3.15 Окно выдачи логин пароля для врачей

### **3.2. Основы физиологии труда и комфортные условия жизнедеятельности деятельности человека**

Характер и организация трудовой деятельности оказывают существенное влияние на изменение функционального состояния организма человека. Многообразные формы трудовой деятельности делятся на физический и умственный труд [18].

Физический труд характеризуется в первую очередь повышенной нагрузкой на опорно-двигательный аппарат и его функциональные системы (сердечнососудистую, нервно-мышечную, дыхательную и др.), обеспечивающие его деятельность. Физический труд, развивая мышечную систему и стимулируя обменные процессы, в то же время имеет ряд отрицательных последствий. Прежде всего это социальная неэффективность физического труда, связанная с низкой его производительностью, необходимостью высокого напряжения физических сил и потребностью в длительном – 50 % рабочего времени – отдыхе.

Умственный труд объединяет работы, связанные с приемом и переработкой информации, требующей преимущественного напряжения сенсорного аппарата, внимания, памяти, а также активизации процессов мышления, эмоциональной сферы. Для данного вида труда характерна гипокинезия, т.е. значительное снижение двигательной активности человека, приводящее к ухудшению реактивности организма и повышению эмоционального напряжения. Гипокинезия является одним из условий формирования сердечнососудистой патологии у лиц умственного труда. Длительная умственная нагрузка оказывает угнетающее влияние на психическую деятельность: ухудшаются функции внимания

(объем, концентрация, переключение), памяти (кратковременной и долговременной), восприятия (появляется большое число ошибок).

В современной трудовой деятельности чисто физический труд не играет существенной роли. В соответствии с существующей физиологической классификацией трудовой деятельности различают формы труда, требующие значительной мышечной активности, механизированные формы труда, формы труда, связанные с полуавтоматическим и автоматическим производством, групповые формы труда (конвейеры), формы труда, связанные с дистанционным управлением, и формы труда интеллектуального (умственного) труда.

Труд преподавателей и медицинских работников отличается постоянными контактами с людьми, повышенной ответственностью, часто дефицитом времени и информации для принятия правильного решения, что обуславливает степень нервно-эмоционального напряжения. Труд учащихся и студентов характеризуется напряжением основных психических функций, таких как память, внимание, восприятие; наличием стрессовых ситуаций (экзамены, зачеты).

Допустимые условия труда характеризуются такими уровнями факторов среды и трудового процесса, которые не превышают установленных гигиеническими нормативами для рабочих мест. Изменения функционального состояния организма восстанавливаются во время регламентированного отдыха или к началу следующей смены, они не должны оказывать неблагоприятное воздействие в ближайшем и отдаленном периоде на здоровье работающего и его потомства. Оптимальный и допустимый классы соответствуют безопасным условиям труда.

Экстремальные условия труда характеризуются такими уровнями производственных факторов, воздействие которых в течение рабочей

смены (или ее части) создает угрозу для жизни, высокий риск возникновения тяжелых форм острых профессиональных поражений.

Эффективность трудовой деятельности человека в значительной степени зависит от предмета и орудий труда, работоспособности организма, организации рабочего места, гигиенических факторов производственной среды.

Работоспособность - величина функциональных возможностей организма человека, характеризующаяся количеством и качеством работы, выполняемой за определенное время. Во время трудовой деятельности работоспособность организма изменяется во времени [18]. Различают три основные фазы сменяющих друг друга состояний человека в процессе трудовой деятельности:

- фаза вработывания, или нарастающей работоспособности; в этот период уровень работоспособности постепенно повышается по сравнению с исходным; в зависимости от характера труда и индивидуальных особенностей человека этот период длится от нескольких минут до 1,5 ч, а при умственном творческом труде - до 2...2,5 ч;

- фаза высокой устойчивости работоспособности; для нее характерно сочетание высоких трудовых показателей с относительной стабильностью или даже некоторым снижением напряженности физиологических функций; продолжительность этой фазы может составлять 2...2,5 ч и более в зависимости от тяжести и напряженности труда;

- фаза снижения работоспособности, характеризующаяся уменьшением функциональных возможностей основных работающих органов человека и сопровождающаяся чувством усталости.

При выполнении работы, требующей значительных усилий и участия крупных мышц, рекомендуются более редкие, но продолжительные 10...12-минутные перерывы.

При выполнении особо тяжелых работ (металлурги, кузнецы и др.) следует сочетать работу в течение 15...20 мин с отдыхом такой продолжительности. При работах, требующих большого нервного напряжения и внимания, быстрых и точных движений рук, целесообразны более частые, но короткие 5...10-минутные перерывы.

В основе благоприятного действия музыки лежит вызываемый ею положительный эмоциональный настрой, необходимый для любого вида работ. Производственная музыка способствует снижению утомляемости, улучшению настроения и здоровья работающих, повышает работоспособность и производительность труда. Однако функциональную музыку не рекомендуется применять при выполнении работ, требующих значительной концентрации внимания (более 70 % рабочего времени), при умственной работе (более 70 % рабочего времени), при большой напряженности выполняемых работ, непостоянных рабочих местах и в неблагоприятных санитарно-гигиенических условиях внешней среды [19].

Эффект психоэмоциональной разгрузки достигается путем эстетического оформления интерьера, использования удобной мебели, позволяющей находиться в удобной расслабленной позе, трансляции специально подобранных музыкальных произведений, насыщения воздуха благотворно действующими отрицательными ионами, приема тонизирующих напитков, имитации в помещении естественно-природного окружения и воспроизведения звуков леса, морского прибоя и др.

## ЗАКЛЮЧЕНИЕ

В данной работе была разработана информационная система центральной больницы Жамбайского района и модуль интерактивных услуг. В программном модуле автоматизированы услуги для территориального населения семейной больницы.

В работе исследовались следующие задачи:

- рассмотрены основные понятия информационных систем;
- изучены и анализированы проектирование информационных систем в образовании;
- обзор инструментальных средств для создания информационных систем;
- разработано интерактивные услуги информационной системы больницы;
- описан порядок пользования информационной системой центральной больницы Жамбайского района
- Рассмотрено комплекс компьютерных программ, предоставляющих пользователям, не владеющим языками программирования, создавать свои компьютерные средства интерактивные услуги через сайта.

Инструментальные средства предоставляют модератору следующие возможности работы:

- готовить разностороннюю информацию;
- формировать сценарий создания определенного интерактивной услуги (ИУ);
- значительно сокращать время на подготовку ИУ и проведения общения через интернет;
- реализовать через созданные ИУ предоставление полезных информации для территориального населения центральной больницы Жамбайского района.

Разработанная ИС интерактивными услугами обсуждено администраторами больницы и решением комиссии ИС размещается в интернет.

## ЛИТЕРАТУРА

1. Постановление Президента Республики Узбекистан “О мерах по дальнейшему внедрению и развитию современных информационно-коммуникационных технологий” от 21 марта 2012 года.  
[http://lex.uz/pages/GetAct.aspx?lact\\_id=1986811](http://lex.uz/pages/GetAct.aspx?lact_id=1986811)
2. Ислам Каримов. О мерах по дальнейшему совершенствованию процедур, связанных с осуществлением предпринимательской деятельности и предоставлением государственных услуг. Ташкент, 15 апреля 2014 года.  
<http://www.press-service.uz/ru/news/4943/>
3. Развивая систему "Электронное правительство": удобство и безопасность.  
<http://www.api.uz/news/info/ict/3285/>
4. Усманов А.И., Корабаев Ж.Ф., Рахматуллаева Ш.К. Информационно-коммуникационные технологии в управлении государством и организацией. – Т.: изд-во «Academiya», 2009
5. Булдакова Т.И. Проектирование информационных систем управления. Уч. пособие ПАГС им. П.А.Столыпина. Саратов, 2007.
6. Глушаков С.В., Жакин И.А., Хачиров Т.С. "Программирование Web-страниц", Изд-во "Фолио", М.: 2003г.
7. Дронов В. А. Самоучитель Macromedia Dreamweaver 8. — СПб.: БХВ-Петербург, 2006. —320 с
8. Дронов В. А. PHP, MySQL и Dreamweaver MX 2004. Разработка интерактивных Web-страниц. — СПб.: БХВ-Петербург, 2005. — 448 с
9. А. Божко «Dreamweaver 4. Базовый курс». – Москва: «ДЕСС КОМ», 2001. – 448 с.
10. Дунаев В. Самоучитель JavaScript, 2-е изд. - СПб.: Питер, 2005. - 395 с.
11. Гудман Д. JavaScript и DHTML. Сборник рецептов. Для профессионалов — СПб.: Питер, 2004.— 523 с.
12. Гудман Д., Моррисон М. JavaScript. Библия пользователя. 5-е издание.: Пер с

- англ. – М.: ООО «И.Д. Вильямс», 2006. – 1184 с.
13. Александр Качанов Букварь по PHP и MySQL, 2000
  14. Dreamweaver 4. Краткий курс Карпов Б, Жданов А. 2001 Питер.
  15. Универсальный Dynamic HTML .,Москва , Санкт-Петербург, Минск 2001г.
  16. А. Кухарчик PHP: обучение на примерах, Минск , 2004.
  17. Қаҳҳоров С., Ҳайитов А.Ғ., Даминов М. Масофавий таълимда маълумотларни бошқариш - Меҳнат ва касб таълими бакалавриат йўналишида муҳандислик фанларини ўқитишнинг долзарб масалалари. Республика илмий-амалий анжумани тезислари. -Наманган, 2005. –Б. 26.
  18. Безопасность жизнедеятельности. Безопасность технологических процессов и производств: Учебное пособие для вузов/ П.П. Кукин, В.Л. Лапин, Е.А. Подгорных. – М. Высш. шк., 1999.
  19. Долин П.А. Основы техники безопасности в электроустановках. М.: Энергоатомиздат, 1984.
  20. <http://www.texnic.ru/books/opis/bgd/bgd001/bgd001.htm>
  21. [www.allbest.ru](http://www.allbest.ru)
  22. [www.ziyonet.ru](http://www.ziyonet.ru)

## Приложение

Index.php fayli

```
<?php include('header.php'); ?>
<!-- -->
<div class="navbar navbar-fixed-top navbar-inverse">
  <div class="navbar-inner">
    <div class="container">
      <a class="btn btn-navbar" data-toggle="collapse" data-target=".nav-
collapse">
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </a>
      <div class="nav-collapse collapse">
        <ul class="nav">
          <li class="active" ><a rel="tooltip" data-placement="bottom"
title="Bosh sahifa" id="home" href="index.php" class="">&nbsp;Bosh sahifa</a></li>
        </ul>
        <form class="navbar-search pull-right">
          <input type="text" class="search-query" placeholder="Qidirish">
        </form>
      </div>
    </div>
  </div>
</div>
<!-- -->
<?php include('dbcon.php'); ?>
<div class="container">
  <div class="margin-top">
    <div class="row">
      <div class="span12">
        <?php include('banner.php'); ?>
      </div>
      <div class="span12">
        <div class="caption_index">FIDOYILIK, TASHABBUSKORLIK – FAOLIYATNING
MUHIM MEZONI</div>
      </div>
      <div class="clearfix"></div>
      <div class="span12">
        <?php include('thumbnail.php'); ?>
      </div>
      <div class="span12">
        <?php// include('content1.php'); ?>
      </div>
      <div class="span12">
        <?php// include('content2.php'); ?>
      </div>
    </div>
  </div>
</div>
```

```

        </div>
    </div>
</div>
<?php include('footer.php') ?>

Bosh oyna header.php
<?php session_start(); ?>
<!DOCTYPE html>
<html>
<head>
    <title>Bosh sahifa</title>
        <link href="img/log.png" rel="icon">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <!-- Bootstrap -->
        <link href="css/bootstrap.css" rel="stylesheet" media="screen">
        <link href="css/bootstrap-responsive.css" rel="stylesheet"
media="screen">
        <link href="css/docs.css" rel="stylesheet" media="screen">
        <link href="css/diapo.css" rel="stylesheet" media="screen">
        <link href="css/font-awesome.css" rel="stylesheet" media="screen">
    <!-- js -->
    <script src="js/jquery-1.7.2.min.js"></script>
    <script src="js/bootstrap.js"></script>
    <script src="js/jquery.hoverdir.js"></script>

        <link rel="stylesheet" type="text/css" href="css/style.css" />
    <noscript>
        <style>
            .da-thumbs li a div {
                top: 0px;
                left: -100%;
                -webkit-transition: all 0.3s ease;
                -moz-transition: all 0.3s ease-in-out;
                -o-transition: all 0.3s ease-in-out;
                -ms-transition: all 0.3s ease-in-out;
                transition: all 0.3s ease-in-out;
            }
            .da-thumbs li a: hover div{
                left: 0px;
            }
        </style>
    </noscript>

    <!--sa calendar-->
    <script type="text/javascript" src="js/datepicker.js"></script>
    <link href="css/datepicker.css" rel="stylesheet" type="text/css" />
    <script type="text/javascript">

```

```

function makeTwoChars(inp) {
    return String(inp).length < 2 ? "0" + inp : inp;
}

function initialiseInputs() {
    // Clear any old values from the inputs (that might be cached by the
    browser after a page reload)
    document.getElementById("sd").value = "";
    document.getElementById("ed").value = "";

    // Add the onchange event handler to the start date input
    datePickerController.addEvent(document.getElementById("sd"),
"change", setReservationDates);
}

var initAttempts = 0;

function setReservationDates(e) {
    // Internet Explorer will not have created the datePickers yet so we
    poll the datePickerController Object using a setTimeout
    // until they become available (a maximum of ten times in case
    something has gone horribly wrong)

    try {
        var sd = datePickerController.getDatePicker("sd");
        var ed = datePickerController.getDatePicker("ed");
    } catch (err) {
        if(initAttempts++ < 10) setTimeout("setReservationDates()",
50);

        return;
    }

    // Check the value of the input is a date of the correct format
    var dt = datePickerController.dateFormat(this.value,
sd.format.charAt(0) == "m");

    // If the input's value cannot be parsed as a valid date then return
    if(dt == 0) return;

    // At this stage we have a valid YYYYMMDD date

    // Grab the value set within the endDate input and parse it using the
    dateFormat method
    // N.B: The second parameter to the dateFormat function, if TRUE,
    tells the function to favour the m-d-y date format
    var edv =
datePickerController.dateFormat(document.getElementById("ed").value,
ed.format.charAt(0) == "m");

```

```

        // Set the low range of the second datePicker to be the date parsed
from the first
        ed.setRangeLow( dt );

        // If theres a value already present within the end date input and
it's smaller than the start date
        // then clear the end date value
        if(edv < dt) {
            document.getElementById("ed").value = "";
        }
    }

    function removeInputEvents() {
        // Remove the onchange event handler set within the function
initialiseInputs
        datePickerController.removeEvent(document.getElementById("sd"),
"change", setReservationDates);
    }

    datePickerController.addEvent(window, 'load', initialiseInputs);
    datePickerController.addEvent(window, 'unload', removeInputEvents);

    //]]>
</script>
</head>
<body>

```

Tizimga kirish (login.php)

```

<?php include('header.php'); ?>
<?php include('navbar.php'); ?>
<?php include('dbcon.php'); ?>
    <div class="container">
        <div class="margin-top">
            <div class="row">
                <div class="span12">
                    
                    <div class="login_sign_up">
                        <a rel="tooltip" data-placement="left" title="Tizimga kirish uchun"
id="login" href="login.php" class="btn btn-info btn-large">&nbsp;Kirish</a>
                        <p><a rel="tooltip" data-placement="bottom" title="Tizimda
ro`yxatdan o`tish uchun" id="signup" href="signup.php">
TIZIMDA RO`YXATDAN O`TISH</a></p>
                    </div>
                    <!-- login -->
                    <?php include('login_modal.php'); ?>
                    <!-- end login -->
                </div>
            </div>
        </div>
    </div>

```

```

        <div class="clearfix"></div>

    </div>
</div>
</div>
<?php include('footer.php') ?>

<?php include('header.php'); ?>
<?php include('navbar.php'); ?>
<?php include('dbcon.php'); ?>
    <div class="container">
        <div class="margin-top">
            <div class="row">
                <div class="span12">
                    <div class="alert alert-danger">Please check your Username and
Password</div>
                    <form method="post" class="form-horizontal">
                        <div class="control-group">
                            <label class="control-label" for="inputPassword">Username</label>
                            <div class="controls">
                                <input type="text" name="username" placeholder="Username" required>
                            </div>
                        </div>
                        <div class="control-group">
                            <label class="control-label" for="inputPassword">Password</label>
                            <div class="controls">
                                <input type="password" name="password" placeholder="Password" required>
                            </div>
                        </div>

                        <div class="control-group">
                            <div class="controls">
                                <button name="submit1" type="submit" class="btn btn-info"><i class="icon-
signin icon-large"></i>&nbsp;Login</button>
                            </div>
                        </div>

                    <?php
if (isset($_POST['submit1'])) {
$username = $_POST['username'];
$password = $_POST['password'];
$query = "SELECT * FROM members WHERE username='\$username' AND password='\$password'";
$result = mysql_query($query) or die(mysql_error());
$num_row = mysql_num_rows($result);
    $row=mysql_fetch_array($result);
    if( $num_row > 0 ) {
        $_SESSION['id']=$row['member_id']; ?>
        <script>
window.location="dashboard.php";

```

```

</script>
<?php }
    else{ ?>
<script>
window.location="login_error.php";
</script>
<?php
}
    }
?>
</form>

</div>
<div class="span12">
<div class="caption_index">we provide the best quality service</div>
</div>
<div class="clearfix"></div>
<div class="span12">
    <?php include('thumbnail.php'); ?>
</div>
<div class="span12">
<?php include('content1.php'); ?>
</div>
<div class="span12">
<?php include('content2.php'); ?>
</div>
</div>
</div>
</div>
<?php include('footer.php') ?>

<div class="alert alert-info">Maydonlarni to`ldiring:</div>
<div class="lgoin_tery">
<form method="post" class="form-horizontal">
    <div class="control-group">
        <label class="control-label" for="inputPassword">Login</label>
        <div class="controls">
            <input type="text" name="username" placeholder="Login" required>
        </div>
    </div>
    <div class="control-group">
        <label class="control-label" for="inputPassword">Parol</label>
        <div class="controls">
            <input type="password" name="password" placeholder="Parol" required>
        </div>
    </div>
    <div class="control-group">

        <div class="controls">

    </div>

```

```

        </div>
        <div class="control-group">
            <div class="controls">
                <button name="submit1" type="submit" class="btn btn-
info">&nbsp;KIRISH</button>
            </div>
        </div>

        <?php
if (isset($_POST['submit1'])) {

$username = $_POST['username'];
$password = $_POST['password'];
$query = "SELECT * FROM members WHERE username='$username' AND password='$password'";
$result = mysql_query($query) or die(mysql_error());
$num_row = mysql_num_rows($result);
    $row=mysql_fetch_array($result);
    if( $num_row > 0 ) {
        $_SESSION['id']=$row['member_id']; ?>
        <script>
        window.location="dasboard.php";
        </script>
        <?php    }
        else{ ?>
            <div class="alert alert-danger"><strong>Xato</strong>&nbsp;Login yoki parol
xato kiritildi.</div>
            <?php
            }
        }
    ?>
</form>
</div>

```

Tizimdan chiqish (logout.php)

```

<?php
session_start();
session_destroy();
header('location:index.php');
?>
<?php include('header.php'); ?>
<?php include('session.php'); ?>
<?php include('dbcon.php'); ?>
<?php include('navbar_dashboard.php'); ?>
    <div class="container">
        <div class="margin-top">
            <div class="row">

                <div class="span3">
                    <ul class="nav nav-tabs nav-stacked">
                        <li class="active">

```

```

        <a href="#">Murojaatlar</a>
    </li>

</ul>
<p><strong>Bugungi sana:</strong></p>
<div class="alert alert-success">

    <?php
    $Today = date('d:m:y');
    $new = date('d.m. Y', strtotime($Today));
    echo $new;
    ?>
</div>
<div class="alert alert-info">Foydalanish bo`yicha yo`riqnom</div>
    <p>Murojrat qilish uchun <b>sana</b> , <b>ma'lumotnoma
turini</b> va <b>vaqt</b> ni
        tanlang.Qabul qilish tugmasini bosing. Hosil bo`lgan
muloqot oynasidan <b>Ha</b> buyrug`i tanlanganda sizning murojaatingiz qabul
qilinadi.</p>

<div class="alert alert-info">Ish kunlari</div>
    <p>Dushanba - Shanba (9:00 to 18:00 )</p>
    <p>Yakshanba dam olish kuni</p>

<div class="alert alert-info">Eslatma</div>
<div class="testimonial_div">
    <p>
    Sayt test rejimida ishlamoqda.
    </p>
</div>
</div>
<div class="span6">

<div class="alert alert-info">Ro`yxat</div>

<table cellpadding="0" cellspacing="0" border="0" class="table
table-bordered" id="example">

    <thead>
    <tr>
        <th>Mening raqamim</th>
        <th>Sana</th>
        <th>Xizmat turi</th>

```

```

        </tr>
    </thead>
    <tbody>

        <?php
            $user_query = mysql_query("select * from schedule where
member_id = '$session_id' ")or die(mysql_error());
            while ($row = mysql_fetch_array($user_query)) {
                $id = $row['id'];
                $member_id = $row['member_id'];
                $service_id = $row['service_id'];
                /* member query */
                $member_query = mysql_query("select * from members where
member_id = '$member_id'")or die(mysql_error());
                $member_row = mysql_fetch_array($member_query);
                /* service query */
                $service_query = mysql_query("select * from service where
service_id = '$service_id' ")or die(mysql_error());
                $service_row = mysql_fetch_array($service_query);
                ?>

                <tr class="del<?php echo $id ?>">
                    <td width="100"><?php echo $row['Number']; ?></td>
                    <td><?php echo $row['date']; ?></td>
                    <td><?php echo $service_row['service_offer']; ?></td>

            </tr>
        <?php } ?>

    </tbody>
</table>

</div>
    <div class="span3">
        
        
        
        <ul class="nav nav-list">
            <div class="alert alert-danger"><li class="nav-
header">E`lonlar</li></div>

```

```

        <?php
        $note_query = mysql_query("select * from note ")or
die(mysql_error());
        $note_count =mysql_num_rows($note_query);
        while($note_row = mysql_fetch_array($note_query)){
        if ($note_count > 0){ ?>

        <li><?php echo $note_row['message'] ?></li>
        <?php
        } }
        ?>
        </ul>
        <br>

        <div class="alert alert-info">Ximat ko`rsatish turi ro`yxati</div>
        <table class="table table-bordered">

                <thead>
                <tr>
                        <th>Xizmat turi</th>

                </tr>
        </thead>
        <tbody>

                <?php $user_query=mysql_query("select * from
service")or die(mysql_error());

                while($row=mysql_fetch_array($user_query)){
                $id=$row['service_id']; ?>
                <tr class="del"><?php echo $id ?>">
                <td><?php echo $row['service_offer']; ?></td>

                <?php } ?>

                </tbody>
        </table>

        </div>
        </div>
        </div>
        <?php include('footer.php') ?>
        <div class="navbar navbar-fixed-top navbar-inverse">
                <div class="navbar-inner">
                        <div class="container">
                                <a class="btn btn-navbar" data-toggle="collapse" data-
target=".nav-collapse">
                                        <span class="icon-bar"></span>

```

```

        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
    </a>
    <div class="nav-collapse collapse">
    <ul class="nav">
    <li class="active" ><a rel="tooltip" data-placement="bottom"
title="Home" id="home" href="index.php" class="">&nbsp;Bosh sahifa</a></li>

        </ul>
        <form class="navbar-search pull-right">
    <input type="text" class="search-query" placeholder="Qidirish">
    </form>

        </div>
    </div>
</div>
</div>
<div class="navbar navbar-fixed-top navbar-inverse">
    <div class="navbar-inner">
        <div class="container">
            <a class="btn btn-navbar" data-toggle="collapse" data-
target=".nav-collapse">
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
            </a>
            <div class="nav-collapse collapse">
            <ul class="nav">
            <?php
            $query=mysql_query("select * from members where
member_id='$session_id'")or die(mysql_error());
            $row=mysql_fetch_array($query);
            ?>
            <li><a href="dashboard.php" class=""></a></li>
            <li class="active" ><a href="dashboard.php" class="">Xush
kelibsiz:&nbsp;&nbsp;&nbsp;<?php echo $row['firstname']. " ".$row['lastname'];
?></a></li>
                <li class="dropdown">
            <a class="dropdown-toggle" data-toggle="dropdown" href="#">
Foydalanvchi
            <b class="caret"></b>
            </a>
            <ul class="dropdown-menu">
            <li><a href="edit_info.php">&nbsp;Ma`lumotlarni tahrirlash</a></li>
            <li><a href="myschedule.php">&nbsp;Mening murojaatlarim</a></li>
            <li><a href="logout.php">Chiqish</a></li>
            </ul>
            </li>

```

```

        </ul>
        <form class="navbar-search pull-right">
<input type="text" class="search-query" placeholder="Qidirish">
</form>

    </div>
</div>
</div>
</div>

<?php include('header.php'); ?>
<!-- -->
<div class="navbar navbar-fixed-top navbar-inverse">
    <div class="navbar-inner">
        <div class="container">
            <a class="btn btn-navbar" data-toggle="collapse" data-target=".nav-
collapse">

                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
            </a>
            <div class="nav-collapse collapse">
                <ul class="nav">
                    <li><a rel="tooltip" data-placement="bottom" title="Home"
id="home" href="index.php" class=""><i class="icon-home icon-
large"></i>&nbsp;Home</a></li>
                    <li class="active"><a rel="tooltip" data-placement="bottom"
title="Services" id="services" href="services.php" class=""><i class="icon-list icon-
large"></i>&nbsp;Services</a></li>
                    <li><a rel="tooltip" data-placement="bottom" title="About Us"
id="aboutus" href="about.php" class=""><i class="icon-info icon-
large"></i>&nbsp;About Us</a></li>
                    <li><a rel="tooltip" data-placement="bottom" title="Contact Us"
id="contactus" href="contact_us.php" class=""><i class="icon-phone icon-
large"></i>&nbsp;Contact US</a></li>

                </ul>
                <form class="navbar-search pull-right">
                    <input type="text" class="search-query" placeholder="Search">
                </form>

            </div>
        </div>
    </div>
</div>
</div>

<!-- -->
<?php include('dbcon.php'); ?>
<div class="container">
    <div class="margin-top">

```

```

<div class="row">
  <div class="span12">
    
    <div class="login_sign_up">
      <a rel="tooltip" data-placement="left" title="Click Here to
Login" id="login" href="login.php" class="btn btn-info btn-large"><i class="icon-
signin icon-large"></i>&nbsp;<a href="login.php">Login</a>
      <p><a rel="tooltip" data-placement="bottom" title="Click Here to
Sign UP" id="signup" href="signup.php">Not a Member? Sign Up Now</a></p>
    </div>
    <!-- login -->
    <?php include('services_content.php'); ?>
    <!-- end login -->
  </div>
  <div class="span12">
    <div class="caption_index">we provide the best quality service</div>
  </div>
  <div class="clearfix"></div>
  <div class="span12">
    <?php include('thumbnail.php'); ?>
  </div>
  <div class="span12">
    <?php include('content1.php'); ?>
  </div>
  <div class="span12">
    <?php include('content2.php'); ?>
  </div>
</div>
</div>
<?php include('footer.php') ?>

<?php
include('dbcon.php');
if (isset($_POST['submit'])){
$firstname=$_POST['firstname'];
$lastname=$_POST['lastname'];
$middlename=$_POST['middlename'];
$age=$_POST['age'];

$address=$_POST['address'];

$contact_no=$_POST['contact_no'];
$username=$_POST['username'];
$password=$_POST['password'];
$password=$_POST['password'];

if($password!=$password){
$a="Parol mos kelmayapti.";

```

```

    }else{
    $a = "";
    }
}
?>
<form method="post">
  <div class="span5">
    <div class="form-horizontal">
      <div class="control-group">
        <label class="control-label" for="inputEmail">F.I.O</label>
        <div class="controls">
          <input type="text" name="firstname" value="<?php if
(isset($_POST['submit']))){echo $firstname;} ?>" placeholder="Ism" required>
          <input type="text" name="lastname" value="<?php if
(isset($_POST['submit']))){echo $lastname;} ?>" placeholder="Familiya" required>
          <input type="text" name="middlename" value="<?php if
(isset($_POST['submit']))){echo $middlename;} ?>" placeholder="Otasining ismi"
required>
        </div>
      </div>

      <div class="control-group">
        <label class="control-label" for="inputPassword">Tug`ilgan yil.</label>
        <div class="controls">
          <input name="age" class="span1" type="text" value="<?php if
(isset($_POST['submit']))){echo $age;} ?>" placeholder="1988" required>
        </div>
      </div>

      <div class="control-group">
        <label class="control-label" for="inputPassword">Login</label>
        <div class="controls">
          <input type="text" name="username" value="<?php if
(isset($_POST['submit']))){echo $username;} ?>" placeholder="Login" required>
        </div>
      </div>

      <div class="control-group">
        <label class="control-label" for="inputPassword">Parol</label>
        <div class="controls">
          <input type="password" name="password" value="<?php if
(isset($_POST['submit']))){echo $password;} ?>" placeholder="Parol">
        </div>
      </div>

      <div class="control-group">
        <div class="controls">
          <button name="submit" type="submit" class="btn btn-
info">&nbsp;OK</button>
        </div>
      </div>

```

```

        </div>
    </div>
</div>

<div class="span6">

<div class="form-horizontal">

    <div class="control-group">
        <label class="control-label" for="inputPassword">Telefon raqam</label>
        <div class="controls">
            <input type="text" name="contact_no" value="<?php if
(isset($_POST['submit']))){echo $contact_no;} ?>"placeholder="998 93 345 53 20"
required>
        </div>
    </div>

    <div class="control-group">
        <label class="control-label" for="inputPassword"> Ish va o`qish
joyi</label>
        <div class="controls">
            <input name="address" type="text" value="<?php if
(isset($_POST['submit']))){echo $email;} ?>" placeholder="" required>
        </div>
    </div>

    <div class="control-group">
        <label class="control-label" for="inputEmail"></label>
        <div class="controls">
            <script type="text/javascript">
                jQuery(document).ready(function() {
                    $('#refresh').tooltip('show');
                    $('#refresh').tooltip('hide');
                })
            </script>
            
            <a href='javascript: refreshCaptcha();'><i data-placement="right"
id="refresh" title="Click to Refresh Code" class="icon-refresh icon-large icon-
spin"></i></a>
            <script language='JavaScript' type='text/javascript'>
                function refreshCaptcha()
                {
                    var img = document.images['captchaimg'];
                    img.src =
img.src.substring(0,img.src.lastIndexOf("?")+"?rand="+Math.random()*1000;
                }
            </script>

```

```

        </div>
    </div>

    <div class="control-group">
    <label class="control-label" for="inputPassword">Kodni kiriting:</label>
    <div class="controls">
    <input id="code" name="code" type="text" placeholder="" required></td>

    <?php

if(isset($_POST['submit']))
{
    $firstname=$_POST['firstname'];
    $lastname=$_POST['lastname'];
    $middlename=$_POST['middlename'];
    $age=$_POST['age'];

    $address=$_POST['address'];

    $contact_no=$_POST['contact_no'];
    $username=$_POST['username'];
    $password=$_POST['password'];
    $cpassword=$_POST['cpassword'];
    if(strcmp($_SESSION['code'], $_POST['code']) != 0)
    {
        ?>
        <span class="label label-important">Kod mos tushmadi.</span>
    <?php
}else if(strcmp($_SESSION['code'], $_POST['code']) == 0 && $password == $cpassword){
    ?>
    <?php
        mysql_query("insert into members
        (firstname,lastname,middlename,age,address,contact_no,username,password)
        values
        ('$firstname','$lastname','$middlename','$age','$address','$contact_no','$username','$password')
        ")or die(mysql_error());?>
    <script type="text/javascript">
    window.location='success.php';
    </script>
    <?php
}else{
    echo " ";
}
?>
    </div>
    </div>

    <div class="control-group">

```

```
        <label class="control-label" for="inputPassword">Parolni qayta
kiriting</label>
        <div class="controls">
            <input type="password" name="cpassword" value="<?php if
(isset($_POST['submit']))){echo $cpassword;} ?>" placeholder="" required>
            <?php if (isset($_POST['submit'])){?> <span class="label label-
important"><?php echo $a; ?></span><?php }?>
            </div>

        </div>

    </div>

</div>

</form>
```