

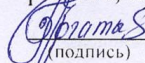
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ МОРДОВСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМ. Н. П. ОГАРЁВА»

Факультет довузовской подготовки и среднего профессионального образования

Выпускающая предметная цикловая комиссия (кафедра) по
общепрофессиональным и специальным (информационно-коммуникационным)
дисциплинам

УТВЕРЖДАЮ

Зам. декана по учебной
работе, преподаватель ВКК


(подпись) В. А. Богатырская

« 7 » _____ 2018 г.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА К ДИПЛОМНОМУ ПРОЕКТУ

на тему Разработка программного модуля «Журнал технического обслуживания
и инвентаризация» информационной системы «Обслуживание технических
средств и оборудования» (на примере ГБУЗ РМ «Поликлиника №4»).

Автор проекта


(подпись)

25.05.18
(дата)

А. А. Камаева

Специальность 090203 Программирование в компьютерных системах

Обозначение дипломного проекта ДП-02069964-090203-07-18

Руководитель проекта

преподаватель ВКК


(подпись)

25.05.18
(дата)

С. Е. Лузгин

Нормоконтролёр

преподаватель ВКК


(подпись)

27.05.18
(дата)

М. Ф. Петянкин

Рецензент

инженер-программист

информационно-аналитического отдела
ГБУЗ РМ «Поликлиника №4»


(подпись)

04.06.18
(дата)

А. В. Казакова

Саранск
2018

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ МОРДОВСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМ. Н. П. ОГАРЁВА»


Факультет довузовской подготовки и среднего профессионального образования

Выпускающая предметная цикловая комиссия (кафедра) по
обще профессиональным и специальным (информационно-коммуникационным)
дисциплинам

УТВЕРЖДАЮ

Зам. декана по учебной

работе, преподаватель ВКК

 В. А. Богатырская

(подпись)

«15» марта 2018 г.

ЗАДАНИЕ НА ДИПЛОМНЫЙ ПРОЕКТ

Студент Камаева Анастасия Андреевна

1 Тема Разработка программного модуля «Журнал технического обслуживания и инвентаризация» информационной системы «Обслуживание технических средств и оборудования» (на примере ГБУЗ РМ «Поликлиника №4»)

Утверждено приказом № 2878–с от 12.04.2018

2 Срок представления работы к защите 24.05.2018

3 Исходные данные для научного исследования (проектирования) технические требования на разработку информационной системы «Журнал технического обслуживания», материалы, полученные из информационно-аналитического отдела ГБУЗ РМ «Поликлиника №4»

Программный модуль должен быть реализован с интуитивно понятным интерфейсом пользователя и выполнять следующие функции:

а) хранить информацию о технических средствах и составе автоматизированных рабочих мест;

б) хранить информацию об обслуживании технических средств и оборудования;

в) выводить данные о технических средствах на печать;

- г) формировать отчёты по запросам пользователя:
- 1) о составе технических средств;
 - 2) о составе автоматизированного рабочего места;
 - 3) осуществление поиска технических средств и автоматизированных рабочих мест;
 - 4) осуществлять поиск записей в журнале технического обслуживания;
 - 5) автоматизация ввода и модификации данных.

4 Содержание выпускной квалификационной работы

4.1 Проектирование программного модуля

4.1.1 Изучение и анализ предметной области

4.1.2 Проектирование сущностей базы данных

4.1.3 Описание методов проектирования, разработки и реализации программного модуля

4.2 Разработка программного модуля

4.2.1 Разработка базы данных

4.2.2 Требование к составу пользователей информационной ситемы

4.3 Реализация программного модуля

4.3.1 Реализация базы данных

4.3.2 Создание интерфейсной части программы

4.3.3 Реализация процедур модуля

4.3.4 Тестирование программного модуля

5 Приложения:

ПРИЛОЖЕНИЕ А (обязательное) Технические требования

ПРИЛОЖЕНИЕ Б (обязательное) Диаграммы базы данных

ПРИЛОЖЕНИЕ В (обязательное) Листинг функций программного модуля

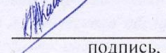
ПРИЛОЖЕНИЕ Г (справочное) Внешний вид форм интерфейса пользователя

Руководитель работы


_____ 15.03.18
подпись, дата

С. Е. Лузгин

Задание принял к исполнению


_____ 15.03.18
подпись, дата

А. А. Камаева

РЕФЕРАТ

Пояснительная записка к дипломному проекту содержит 87 страниц, 4 рисунка, 19 использованных источников, 4 приложения и 9 таблиц.





ИНФОРМАЦИОННАЯ СИСТЕМА, ПРОГРАММНЫЙ МОДУЛЬ, ТЕХНИЧЕСКОЕ СРЕДСТВО, ТЕХНИЧЕСКОЕ ОБСЛУЖИВАНИЕ, АВТОМАТИЗИРОВАННОЕ РАБОЧЕЕ МЕСТО, ОБОРУДОВАНИЕ, УЧЁТ, ИНВЕНТАРИЗАЦИЯ, ПОЛИКЛИНИКА.

Основной целью проекта является автоматизация процесса учёта оборудования и подачи заявок клиентов на обслуживание технических средств.

Основная цель дипломного проекта на тему «Разработка программного модуля «Журнал технического обслуживания и инвентаризация» информационной системы «Обслуживание технических средств и оборудования»» – создание программного модуля для автоматизации процесса инвентаризации технических средств, составления автоматизированных рабочих мест и ведения журнала технического обслуживания.

В процессе работы проводились: изучение методики учёта технических средств и их последующего обслуживания в ГБУЗ РМ «Поликлиника №4», разработка и реализация базы данных под управлением СУБД MySQL версии 5.6, создание программного модуля на скриптовом языке PHP версии 5.5. При проектировании использовались знания и навыки, полученные при изучении профессиональных модулей «Разработка программных модулей программного обеспечения для компьютерных систем», «Разработка и администрирование баз данных», «Участие в интеграции программных модулей».

В результате проведенной работы получен программный модуль «Журнал технического обслуживания и инвентаризация», необходимый для

ДП-02069964-090203-07-18								
Изм.	Лист	№ докум.	Подпись	Дата	Разработка программного модуля «Журнал технического обслуживания и инвентаризация» информационной системы «Обслуживание технических средств и оборудования» (на примере ГБУЗ РМ «Поликлиника №4»)	Лит.	Лист	Листов
Разраб.	Камаева			25.05		Д	4	87
Провер.	Лузгин			25.05				
Н. Контр.	Петянкин			27.05				
Утверд.	Богатырская			07.06				
					ФДП и СПО, д/о, ПВКС, 409			

работы информационной системы «Обслуживание технических средств и оборудования».

Степень внедрения – частичная.

Область применения – программный продукт может быть использован в составе информационной системы, предназначенной для инвентаризации технических средств и учета состава автоматизированных рабочих мест в государственном бюджетном учреждении здравоохранения Республики Мордовия «Поликлиника №4».

Эффективность – автоматизация работ служб ремонта и технического обслуживания организаций и предприятий.

					<i>ДП-02069964-090203-07-18</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		5

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	7
1 Проектирование программного модуля	13
1.1 Изучение и анализ предметной области	13
1.2 Проектирование сущностей базы данных	15
1.3 Описание методов проектирования, разработки и реализации программного модуля	20
2 Разработка программного модуля	25
2.1 Разработка базы данных	25
2.2 Требование к составу пользователей информационной системы	33
3 Реализация программного модуля	35
3.1 Реализация базы данных	35
3.2 Создание интерфейсной части модуля	37
3.3 Реализация процедур модуля	41
3.4 Тестирование программного модуля	46
ЗАКЛЮЧЕНИЕ	52
СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ	53
ПРИЛОЖЕНИЕ А (обязательное) Технические требования	56
ПРИЛОЖЕНИЕ Б (обязательное) Диаграммы базы данных	59
ПРИЛОЖЕНИЕ В (обязательное) Листинг функций программного модуля	61
ПРИЛОЖЕНИЕ Г (справочное) Внешний вид форм интерфейса пользователя	85

ВВЕДЕНИЕ

Одной из особенностей современного мира является тенденция количественного роста устройств, позволяющих автоматизировать различные виды человеческой деятельности. Однако эксплуатация большого числа различных механизмов, электронных и электромеханических устройств, программного обеспечения и автоматизированных информационных систем приводит к потребности технического обслуживания и необходимости проведения плановых и внеплановых ремонтных работ, осуществляемых инженерами, механиками, электриками, и другими специалистами. Их первостепенной задачей является своевременное оказание технической помощи, технического обслуживания и документирования выполненных работ. Для решения проблем в организации обслуживания необходима автоматизация процесса своевременного получения заявок на ремонт технических средств, выявление неисправностей, которые происходят с конкретным видом техники, планирование технического обслуживания, ведения всей необходимой документации и истории выполненных работ.

К тому же, на производстве, в организациях и на предприятиях требуется своевременно получать фактическую информацию о состоянии и нахождении технических средств, следовательно, и оказывать оперативную помощь пользователям при эксплуатации оборудования.

Автоматизация процесса работы с данными позволяет эффективно манипулировать информацией, предоставляя доступ одновременно нескольким пользователям. Это позволит специалисту мобильно выявить проблему, своевременно получить полную необходимую информацию и произвести требуемую работу с оборудованием.

Автоматизированные информационные системы, основанные на концепции интеграции данных, характеризуются огромными объемами

					<i>ДП-02069964-090203-07-18</i>	<i>Лист</i>
						7
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		

хранимых данных, сложной организацией и необходимостью удовлетворять разнообразные требования многочисленных пользователей.

Технические средства, являющиеся компонентами рабочих мест, постоянно мигрируют, выходят из строя, передаются на другие участки предприятия и организации. Концепция постоянно меняющихся данных о перемещениях оборудования приводит к значительным трудностям их технического учёта, нахождения, отслеживания их состояния. Во избежание возникающих на этой почве проблем в поликлинике было принято решение о создании автоматизированной информационной системы, позволяющей оперативно получать актуальную информацию о состоянии технических средств.

Целью дипломного проекта является создание программного модуля «Журнал технического обслуживания и инвентаризация» информационной системы «Обслуживание технических средств и оборудования». Программный модуль предоставляет возможность добавления данных о введении в эксплуатацию новой техники, просмотра и редактирования данных о технических средствах и их технического обслуживания. При разработке программного модуля необходимо изучить предметную область, выявить основы бизнес-логики процессов автоматизации и учёта материальных средств.

Для достижения поставленной цели решались следующие задачи:

- а) изучить и проанализировать предметную область на основе документов и материалов, предоставленных ГБУЗ РМ «Поликлиника №4»;
- б) разработать инфологическую модель данных инвентаризации технических средств;
- в) реализовать базу данных средствами системы управления базы данных MySQL с использованием языка SQL на основе инфологической модели данных;
- г) спроектировать и реализовать интерфейс;
- д) создать программный модуль авторизации пользователей;

									Лист
									8
Изм.	Лист	№ докум.	Подпись	Дата	ДП-02069964-090203-07-18				

- е) разработать функции обработки данных;
- ж) скомпоновать модуль инвентаризации технических средств;
- з) разработать тестовые наборы и провести тестирование созданного программного модуля.

Аналогами разрабатываемой информационной системы являются такие программные продукты, как IT Invent и JIRA.

Программная система IT Invent позволяет вести учёт и инвентаризацию оборудования на предприятии, учёт офисной техники, принтеров и перезаправки картриджей, учёт оргтехники, учёт программного обеспечения, комплектующих, расходных материалов и любой другой техники, а также мебели и другого инвентаря. Это электронный журнал учета оборудования. Инвентаризация проводится как в рамках одной организации, так и в нескольких филиалах. Помимо этого, есть сетевое сканирование и инвентаризация компьютеров в сети с автоматическим сбором данных. Программа для сканирования сети создаёт оборудование в базе на основе собранных данных. При помощи системы IT Invent инвентаризация сети и активного сетевого оборудования – простая задача. Данная система поможет сотрудникам информационных и производственно-технических отделов иметь точную информацию о состоянии технического оборудования. Удобный и функциональный интерфейс позволяет мгновенно получать доступ ко всем параметрам и характеристикам учитываемого технического оборудования и отслеживать производимые коррективы. Программы для системного администратора обязательно включают в себя систему инвентаризации. Обладая централизованной информацией, программа инвентаризации компьютеров IT Invent дает возможность планировать и учитывать закупки, замены, ремонты, обслуживания, работы и списания компьютеров и любого другого оборудования [12].

Однако программа IT Invent ограничивается реализацией учёта

					<i>ДП-02069964-090203-07-18</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		9

технических средств, не предоставляя возможности вести учёт и обслуживание поступающих заявок на ремонт оборудования.

Требования на реализацию функции обработки заявок основаны на документах и материалах, предоставленных ГБУЗ РМ «Поликлиника №4». В поликлинике имеется технический и информационный отделы, в которых сотрудники занимаются заполнением журналов учёта технических средств, ремонтом оборудования и введения журналов учёта заявок на ремонт. При внесении данных в журналы уходит большое количество времени и бывает, когда сотрудник предприятия либо не успевает обработать все заявки в указанный срок, либо попросту может забыть про какую-либо заявку. Эти две проблемы могут привести к тому, что будут приходить жалобы не только пользователей технических средств, но и пациентов и при этом репутация организации может пострадать. Чтобы решить эту проблему, возникла необходимость автоматизировать как процесс учёта технических средств, так и учёта заявок клиента на ремонт аппаратуры.

Системой, подходящей для отслеживания ошибок и управления проектом в компании любого размера, является JIRA. Это инструмент для всех сотрудников в команде и руководителей проектов.

JIRA помогает оперативно обмениваться информацией и легко вовлекать разных работников в проекты и задачи, мониторить пользовательские ошибки при работе с программными продуктами, фиксировать своевременное соблюдение работы и назначать определённые задачи в соответствии с определяемой эффективностью участников, работать вместе с коллегами с помощью инструментов совместного редактирования файлов, а также отслеживать прогресс и обновление каждой задачи команды.

Динамичные инструменты системы для управления проектами JIRA дают возможность руководителям обнаружить препятствия, которые не дают команде работать эффективнее, принимать целенаправленные действия по их устранению и определять области улучшения рабочего процесса.

					<i>ДП-02069964-090203-07-18</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		10

Базовые возможности JIRA включают в себя организацию результативной работы с проектами, отслеживание эффективности рабочего процесса, его планирование, совместную работу сотрудников по проекту, а также интеграцию с различными разработками и дополнениями Altassian и другими разработчиками [11].

Несмотря на все возможности JIRA, данная система имеет один существенный недостаток: система JIRA не предназначена для инвентаризации и учёта технических средств.

В связи с тем, что на данный момент не существует системы, совмещающей в себе и модуль инвентаризации и модуль обработки заявок, возникает необходимость в создании подобного программного обеспечения.

На момент наиболее активного пользования Всемирной информационной сетью актуально создание информационных систем по типу интерактивного сайта. Данный продукт легок и прост в эксплуатации, понятен пользователю, его легко внедрить, а также модернизировать по требованию сотрудников.

Для использования базы данных, ее интеграции в информационную систему и возможного дальнейшего импорта в другие системы, построение физической модели будет осуществлено командами и предложениями языка структурированных запросов SQL. Язык SQL является основой многих СУБД, так как отвечает за физическое структурирование и запись данных на диск, а также за чтение данных с диска, позволяет принимать SQL-запросы от других компонентов СУБД и пользовательских приложений. Таким образом, SQL – мощный инструмент, который обеспечивает пользователям, программам и вычислительным системам доступ к информации, содержащейся в реляционных базах данных [15].

Работы по созданию физической модели и реализации базы данных объективнее всего осуществлять, используя систему управления базами данных MySQL. Преимущества системы в том, что она свободно распространяется,

					<i>ДП-02069964-090203-07-18</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		11

обладает большой гибкостью, а также используется широким кругом пользователей.

Администрирование базы данных будет осуществляться с помощью веб-интерфейса для работы с системой управления базы данных MySQL PHPMyAdmin. Данное приложение позволяет управлять базой данных без ввода дополнительных команд SQL, просматривать содержимое таблиц и предоставляет другие функции управления посредством удобного интерфейса.

Для создания программного модуля выбран PHP – скриптовый язык программирования, который отлично подходит для генерации HTML-страниц на веб-сервере и работы с базами данных, так как PHP входит в LAMP (Linux, Apache, MariaDB / MySQL, PHP комплекс серверного программного обеспечения), в настоящее время он поддерживается почти всеми хостинг провайдерами.

Выбор вышеперечисленных технологий основан на их высоком распространении, с целью разработки достаточно гибкой системы с возможностью интеграции в разные информационные среды.

К основной цели выпускной квалификационной работы относится автоматизация работы сотрудников технического и информационного отдела по обслуживанию технических средств.

1 Проектирование программного модуля

1.1 Изучение и анализ предметной области

Сложная электронная, электронно-механическая и вычислительная техника применяется сегодня на каждом предприятии. В теории бухгалтерского учёта любое техническое устройство понимается как основное средство. Как и прочие основные средства, она подлежит обслуживанию, учёту и инвентаризации. Эта процедура имеет отличительные черты. Порядок проведения описи имущества регламентируют: Закон о бухгалтерском учёте № 402-ФЗ от 06.12.2011, Методические рекомендации, утв. Приказом Министерства финансов № 49 от 13.06.1995, Положение по ведению бухгалтерского учёта, утв. Приказом Министерства финансов № 34н от 29.07.1998.

Инвентаризация – это один из основных методов контроля над сохранностью активов. Она представляет собой проверку наличия имущества и обязательств компании на определенную дату.

Проверка наличия и состояния технических средств предполагает опись относящихся к ней комплектующих. Кроме того, необходимо следить за обновлениями программного обеспечения вычислительной техники и своевременно осуществлять активацию специализированных приложений.

Опись технических средств преследует следующие цели:

- а) контроль сохранности имущества;
- б) определение соответствия фактического наличия данным бухгалтерского учёта;
- в) оптимизация расходов на закупку ЭВМ;
- г) выявление неиспользуемых объектов;
- д) формирование перечня на списание;

- е) контроль изменения стоимости имущества;
- ж) поддержание дисциплины ответственных за технику лиц.

Ежегодно Министерством здравоохранения Российской Федерации составляются планы внедрения информационных технологий и совершенствования медицинской информационной системы.

Для реализации соответствующих приказов в рамках трудовой деятельности составляют автоматизированные рабочие места, представляющие собой совокупности технических средств.

Рабочее место является неотъемлемой частью производственной структуры учреждения. В виду этого организация рабочего места выступает в качестве составной части организации труда. Под рабочим местом понимается часть пространства или зона осуществления трудовых действий, оборудованная необходимыми орудиями труда, на котором выполняется порученная функция (задание, работа) исполнителем или группой исполнителей [5].

Особое внимание стоит уделить автоматизированным рабочим местам. Понятие автоматизированного рабочего места (АРМ) можно раскрыть как совокупность информационных, программных и технических ресурсов, обеспечивающих конечному пользователю обработку данных и автоматизацию функций работника.

Поскольку любое рабочее место предназначено для выполнения работником трудовых обязанностей, оно должно быть оснащено определенным перечнем оборудования и средств труда, необходимых для выполнения трудовых обязанностей работником [9].

Под оснащением рабочего места понимается обеспечение его всеми необходимыми средствами, при помощи которых создаются условия для эффективного выполнения работником своих профессиональных обязанностей. Выделяют следующие виды оснащений рабочих мест [16]:

- а) основное технологическое оборудование, используемое для выполнения основной работы на данном рабочем месте;

- б) вспомогательное оборудование;
- в) технологическая оснастка;
- г) рабочая документация и специализированная литература;
- д) организационная оснастка;
- е) средства обеспечения безопасности условий труда;
- ж) осветительное оборудование;
- з) средства связи;
- и) рабочая тара;
- к) хозяйственные средства для поддержания на рабочем месте чистоты;
- л) средства для поддержания на рабочем месте нормального микроклимата.

Информатизация в области здравоохранения ориентирована на оказание медицинской помощи пациентам. С этой целью создаются соответствующие информационные системы, хранящие большое количество данных. Это требует мощной аппаратной поддержки. Однако любое оборудование может морально устаревать, выходить из строя или мигрировать. Поэтому не менее значимым элементом организации труда, наряду с формированием рабочих мест является построение обслуживания этих рабочих мест.

Однако информационной системы, обеспечивающей не только инвентаризацию технических средств и комплектацию автоматизированных рабочих мест, но и обработку заявок на ремонт оборудования на данный момент не существует.

1.2 Проектирование сущностей базы данных

На данном этапе организация контроля наличия и состояния технических

					<i>ДП-02069964-090203-07-18</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		15

средств, используемых в учреждениях здравоохранения, находится в поле ответственности бухгалтерии и технических служб.

Сотрудники бухгалтерии вручную ведут журналы инвентаризации компьютерной и медицинской техники, данные сумм и баланса. Так как информационная система, в состав которой входит модуль учёта технических средств не включает финансового учета, а только технический учёт, то информация о финансах не нужна. Из данных, учитываемых сотрудниками бухгалтерии актуальны такие идентификационные данные, как инвентарный и серийный номер. В связи с тем, что многие сотрудники бухгалтерии недостаточно компетентны в области аппаратного обеспечения, часто случаются ошибки в учете технических средств, когда наименование технического средства путается с названием автоматизированного рабочего места и наименование комплектующего трактуется как техническое средство. Это приводит к недопониманию между сотрудниками различных служб учреждения, материально-ответственными лицами и технической службы.

Техническая служба следит за состоянием всех технических средств. Для этого используются журналы технического обслуживания, отражающие информацию об устройстве, его свойствах и действий, производимых с ним. Из этого следует, что основным объектом данных разрабатываемого модуля будет являться *«техническое средство»*.

Объект данных «техническое средство» содержит идентификационную информацию, а также характеристики и свойства оборудования. В результате анализа оборотных ведомостей бухгалтерии и журналов технического обслуживания можно выделить следующие основные свойства объекта:

- а) наименование технического средства;
- б) модель технического средства;
- в) серийный номер;
- г) инвентарный номер;
- д) год выпуска;

- е) дата ввода в эксплуатацию;
- ж) технические характеристики;
- з) состояние технического средства.

Исходя из анализа свойств объектов, составлена инфологическая модель, отображающая реальный мир в некоторые понятные человеку концепции, полностью независимые от параметров среды хранения данных. Существует множество подходов к построению таких моделей: графовые модели, семантические сети, модель «сущность-связь» [19]. Наиболее популярной является модель «сущность-связь» или называемая ER-моделью (от англ. Entity-Relationship, т.е. сущность-связь). Данная модель была предложена Питером Ченом, им же разработана и самая популярная графическая нотация для модели.

Используя правила построения ER-модели в нотации Питера Чена, создаётся модель объекта «техническое средство» представленная на рисунке 1.



Рисунок 1 – Модель сущности «техническое средство»

В бухгалтерском учёте автоматизированные рабочие места не учитываются. Однако данный объект необходим для автоматизации в модуле учёта технических средств. Данные об автоматизированных рабочих местах врачей хранятся в информационно-аналитическом отделе поликлиники, а также у инженеров медицинской техники как набор компонентов из технических

средств.

Исходя из данных, полученных при изучении автоматизированных рабочих мест врачей и специалистов поликлиники, выделим следующие свойства объекта «автоматизированное рабочее место»:

- а) компоненты (экземпляры объекта «технические средства»);
- б) местоположение;
- в) ответственное лицо;
- г) дата реализации.

Объект «автоматизированное рабочее место» имеет отношения с объектом «техническое средство», так как автоматизированные рабочие места состояются именно из технических средств. Используя правила построения ER-модели в нотации Питера Чена, создаётся модель объекта «автоматизированное рабочее место» представленная на рисунке 2.

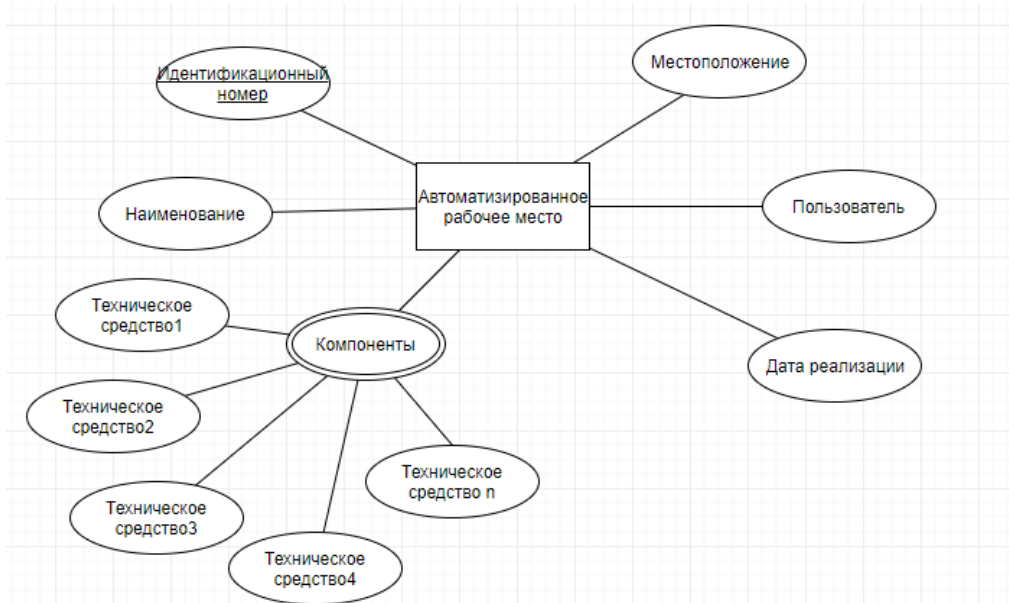


Рисунок 2 – Модель сущности «автоматизированное рабочее место»

Каждый сотрудник при приеме на работу подписывает с организацией трудовой договор. Трудовым договором или заключаемыми в письменной форме соглашениями, прилагаемыми к нему, может конкретизироваться материальная ответственность сторон этого договора [17].

Материальная ответственность стороны трудового договора наступает за ущерб, причиненный другой стороне этого договора в результате ее виновного противоправного поведения (действий или бездействия), если иное не предусмотрено настоящим Кодексом или иными федеральными законами [18].

Работники поликлиники имеют в распоряжении персональные компьютеры, медицинское оборудование и аналогичные технические средства. При заключении договора работодатель закрепляет за сотрудниками автоматизированные рабочие места и определяет материальную ответственность за предоставляемое оборудование.

Исходя из данных о сотрудниках поликлиники, выделим следующие свойства объекта *материально-ответственные лица*.

- а) ФИО;
- б) должность;
- в) прикрепленное автоматизированное рабочее место (экземпляры объекта «автоматизированное рабочее место»).

ER-диаграмма, отражающая все связи между объектами, представлена в приложении Б. Используя правила построения ER-модели в нотации Питера Чена, создаётся модель объекта «материально-ответственные лица» представленная на рисунке 3.

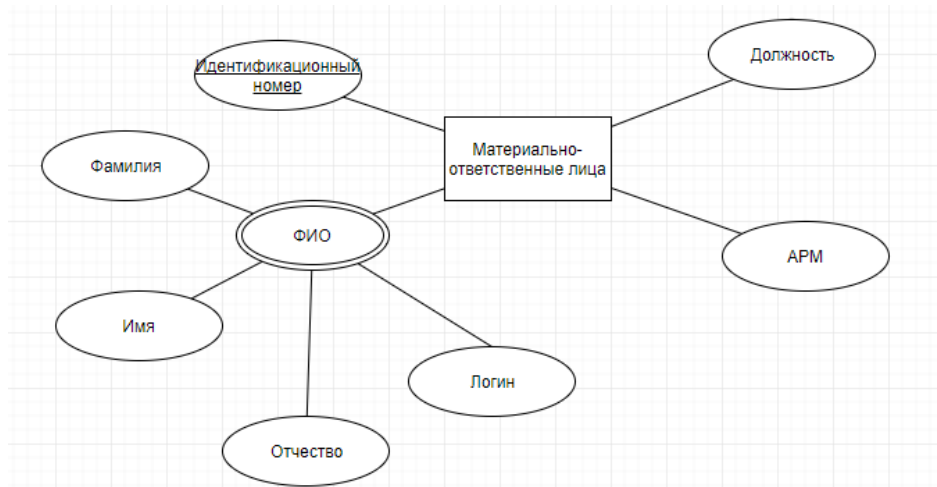


Рисунок 3 – Модель сущности «материально-ответственные лица»

«Местоположение» является словарем значений филиалов. Атрибуты:

- а) филиал;
- б) адрес.

«Статус» является словарем значений статуса работоспособности технических средств и автоматизированных рабочих мест. Имеет атрибут:

- а) статус.

«Параметры технических средств». Является словарем всех параметров различного рода технических средств. Состоит из атрибутов:

- а) наименование технического средства;
- б) набор параметров.

1.3 Описание методов проектирования, разработки и реализации программного модуля

При изучении предметной области одним из основных способов представления информации являются её графические представления. Для наглядного описания сущностей базы данных использовалось приложение «Gliffy Diagram». Это бесплатное приложение отлично подходит для создания диаграмм, моделей или блок-схем. Данное приложение доступно в режиме онлайн на сайте.

В основе проекта лежат системы разработки, созданные по GPL лицензии и предоставляемые в общественный доступ бесплатно.

Основой любой системы учёта является база данных, хранящая в себе необходимую информацию. Для работы с базой данных была выбрана СУБД MySQL Community Server версия программы: 5.6.

MySQL является свободной реляционной системой управления базами данных. Данная СУБД является решением для создания и администрирования

										Лист
										20
Изм.	Лист	№ докум.	Подпись	Дата	ДП-02069964-090203-07-18					

приложений любого размера. MySQL является частью LAMP, представляющего собой комплекс серверного программного обеспечения сервера Linux, Apache, MariaDB / MySQL, PHP. В дистрибутив MySQL входит библиотека внутреннего сервера, позволяющая включать систему в автономные программы, однако в большинстве случаев она используется в качестве сервера, к которому обращаются локальные или удалённые клиенты.

Для управления программным продуктом был выбран сервер Apache – это HTTP сервер, обладающий высокой надежностью и гибкостью.

Для разработки программного продукта в среде Windows использовалась портативная серверная платформа и программная среда Open Server Panel. Программный комплекс имеет богатый набор серверного программного обеспечения, удобный, многофункциональный продуманный интерфейс, обладает мощными возможностями по администрированию и настройке компонентов. Платформа широко используется с целью разработки, отладки и тестирования веб-проектов, а также для предоставления веб-сервисов в локальных сетях [10].

Для графического описания интерфейса пользователя использовалось объектно-ориентированное средство проектирования Rational Rose, являющееся реализацией CASE-технологии. Понятие CASE-технология расшифровывается как Computer-Aided Software/System Engineering – разработка систем (или программного обеспечения) с помощью компьютера. CASE-технологии служат для проектирования программ, приложений или систем, абстрагируясь от реализации проектов на конкретном языке и представляют собой совокупность методологий анализа, проектирования, разработки, кодирования, тестирования и сопровождения сложных систем программного обеспечения.

Создание программного модуля ведения журнала технического обслуживания и инвентаризации удобнее реализовать при помощи приложения PHPMyAdmin, используя язык разметки HTML в совокупности с языком web-программирования PHP.

					<i>ДП-02069964-090203-07-18</i>	<i>Лист</i>
						21
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		

Для написания сценариев удобнее использовать свободный текстовый редактор Notepad++. Безусловным преимуществом данного текстового редактора является подсветка синтаксиса большого количества языков программирования и разметки, а также возможность открытия большого числа форматов. Есть возможность расширить базовую функциональность программы за счёт плагинов или сторонних модулей, таких как препроцессоры или компиляторы.

Для упрощения создания кода немаловажно знать функционал текстового редактора. Базовые возможности программы Notepad++ заявлены на официальном сайте [13]:

- а) подсветка синтаксиса;
- б) сворачивание кода;
- в) автодополнение и автоматическое закрытие скобок и тэгов (если активировано);
- г) закладки;
- д) регулярные выражения для поиска и замены;
- е) запись и воспроизведение макросов;
- ж) сравнение файлов;
- з) менеджер проектов;
- и) карта документа;
- к) переопределение любых горячих клавиш;
- л) резервное копирование сохраняемых файлов (включается в настройках);
- м) трансформация текста при помощи подключённого плагина TextFX;
- н) поддержка и конвертирование кодировок ANSI, UTF-8 и UCS-2;
- о) блоковое выделение текста, одновременное выделение нескольких разных мест (с Ctrl).

Для администрирования системы управления базами данных MySQL использовалось веб-приложение с открытым кодом, написанное на языке PHP –

PHPMyAdmin. Данное приложение позволяет администрировать сервера MySQL, исполнять SQL-запросы и просматривать информацию из баз данных. PHPMyAdmin пользуется популярностью по причине того, что данное приложение позволяет управлять системой управления базами данных без непосредственного ввода SQL-запросов, предоставляя разработчикам удобный интерфейс. Широкая распространённость связана с тем, что российские провайдеры используют PHPMyAdmin для предоставления своим клиентам возможности администрирования выделенных баз данных.

В качестве основного языка разработки используется скриптовый язык общего назначения PHP. Изначальная этимология слова связана с аббревиатурой Personal Home Page Tools, что с английского переводится «Инструменты для создания персональных веб-страниц». В настоящее время данный термин расшифровывается как Hypertext Preprocessor, что на русском звучит как «препроцессор гипертекста», широко применяемый для разработки веб-приложений. В настоящее время поддерживается подавляющим большинством хостинг-провайдеров и является одним из лидеров среди языков, применяющихся для создания динамических веб-сайтов [14].

Для создания страниц использовался стандартизированный язык разметки документов во Всемирной паутине HTML. Аббревиатура расшифровывается как HyperText Markup Language, то есть «язык гипертекстовой разметки». Большинство веб-страниц содержат описание разметки на языке HTML (или XHTML). Язык HTML интерпретируется браузерами; полученный в результате интерпретации форматированный текст отображается на экране монитора компьютера или мобильного устройства [14].

Для оформления дизайна страниц используется CSS формальный язык описания внешнего вида документа, написанного с использованием языка разметки. С английского языка переводится как «каскадные таблицы стилей» – Cascading Style Sheets [14].

Выбор вышеперечисленных технологий основан на их высоком распространении, с целью разработки достаточно гибкой системы с возможностью интеграции в разные информационные системы.

					<i>ДП-02069964-090203-07-18</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		24

2 Разработка программного модуля

2.1 Разработка базы данных

На основе данных, предоставленных ГБУЗ РМ «Поликлиника №4» был произведен анализ предметной области, заключающийся в определении объектов базы данных и разработке инфологической модели базы данных, отраженной в приложении Б. Инфологическая модель не имеет ориентации на какую-либо конкретную систему управления базами данных и модель данных. Инфологическая модель, представленная в приложении Б, является ER-диаграммой, разработанной в соответствии с нотацией Питера Чена.

Следующим этапом проектирования является логическое проектирование, заключающееся в создании схемы базы данных на основе реляционной модели данных. При разработке схемы базы данных наиважнейшей частью является нормализация, представляющая собой процесс преобразования базы данных к виду, который соответствует нормальным формам. Процесс нормализации позволяет избегать аномалий данных. Под аномалиями в данном контексте понимаются структурные и логические проблемы данных. В случае когда существует некоторое количество одинаковых записей в таблице, присутствует риск нарушения целостности данных при обновлении таблицы. При осуществлении нормализации решаются подобные проблемы, так как структура данного процесса предполагает определение связей между данными, исключая необходимость в повторяющихся записях.

Впервые Эдгар Кодд ввёл понятие нормальной формы при создании реляционной модели базы данных. Основное назначение нормальных форм – приведение структуры базы данных к виду, обеспечивающему минимальную избыточность. Устранение избыточности производится за счёт декомпозиции

										Лист
										25
Изм.	Лист	№ докум.	Подпись	Дата	ДП-02069964-090203-07-18					

отношений (таблиц) таким образом, чтобы в каждом отношении хранились только первичные факты (то есть факты, не выводимые из других хранимых фактов). Таким образом, нормализация не имеет целью уменьшение или увеличение производительности работы, или же уменьшение или увеличение объёма базы данных. Конечной целью нормализации является уменьшение потенциальной противоречивости хранимой в базе данных информации [3].

При разработке первой нормальной формы необходимо реализовать уникальность каждого поля таблицы, а точнее каждый составной элемент представить отдельным и убрать дублирующиеся данные. Правила организации первой нормальной формы гласят, что все таблицы должны быть плоскими. Плоской является таблица, в которой на пересечении каждой строки и каждого столбца содержится только одно значение.

В объекте «техническое средство» имеются два многозначных атрибута – «идентификационные номера» и «параметры». В качестве идентификационных номеров технических средств рассматриваются инвентарный и серийный номера. Кроме того, каждое техническое средство имеет ряд характеристик, описывающих его свойства. Для описания свойств технических средств достаточно рассмотреть девять характеристик. Структура объекта «техническое средство» представлена в таблице 1.

Таблица 1 – Структура объекта «техническое средство»

ТЕХНИЧЕСКОЕ СРЕДСТВО / TECHNICAL_MEAN	
СВОЙСТВО ОБЪЕКТА И ЕГО ОПИСАНИЕ	ИМЯ АТТРИБУТА
Наименование технического средства	name
Модель технического средства	model
Серийный номер	serial_number
Инвентарный номер	inventory_number
Год выпуска	release_year

составления автоматизированного рабочего места определяем двадцать компонентов. Структура объекта «автоматизированное рабочее место» представлена в таблице 3.

Таблица 3 – Структура объекта «автоматизированное рабочее место»

АВТОМАТИЗИРОВАННОЕ РАБОЧЕЕ МЕСТО / WORKSTATION	
СВОЙСТВО ОБЪЕКТА И ЕГО ОПИСАНИЕ	ИМЯ АТТРИБУТА
АРМ	workstation
Составляющие	component1
Составляющие	component2
Составляющие	component3
Составляющие	component4
Составляющие	component5
Составляющие	component6
Составляющие	component7
Составляющие	component8
Составляющие	component9
Составляющие	component10
Составляющие	component11
Составляющие	component12
Составляющие	component13
Составляющие	component14
Составляющие	component15
Составляющие	component16
Составляющие	component17

Окончание таблицы 3

Составляющие	component18
Составляющие	component19
Составляющие	component20
Идентификатор филиала	id_location
Идентификатор пользователя	id_user
Дата выпуска в пользование	data_release
Дата изъятия из пользования	data_of_destruction
Статус	status

Кроме описанных при изучении предметной области справочников «местоположение», «статус» и «параметры технических средств» необходимо создать справочник пользователей системы, который будет использоваться при авторизации пользователей. Структура справочников представленная в таблицах 4, 5, 6, 7.

Таблица 4 – Структура справочника «местоположение»

МЕСТОПОЛОЖЕНИЕ / LOCATION	
СВОЙСТВО ОБЪЕКТА И ЕГО ОПИСАНИЕ	ИМЯ АТТРИБУТА
Филиал, корпус	branch
Адрес	address

Таблица 5 – Структура справочника «статус»

СТАТУС / STATUS_MAINTENANCE	
СВОЙСТВО ОБЪЕКТА И ЕГО ОПИСАНИЕ	ИМЯ АТТРИБУТА
Статус	status

Таблица 6 – Структура справочника «параметры технических средств»

ПАРАМЕТРЫ ТЕХНИЧЕСКИХ СРЕДСТВ / TS	
СВОЙСТВО ОБЪЕКТА И ЕГО ОПИСАНИЕ	ИМЯ АТТРИБУТА
Наименование технического средства	id_workstation
Параметр1	option1
Параметр2	option2
Параметр3	option3
Параметр4	option4
Параметр5	option5
Параметр6	option6
Параметр7	option7
Параметр8	option8
Параметр9	option9

Таблица 7 – Структура справочника «пользователи системы»

ПОЛЬЗОВАТЕЛИ СИСТЕМЫ / USERS	
СВОЙСТВО ОБЪЕКТА И ЕГО ОПИСАНИЕ	ИМЯ АТТРИБУТА
Логин	login
Пароль	password
Флаг	flag

Правила построения второй нормальной формы требуют, чтобы была обеспечена полная зависимость столбцов, не являющихся ключевыми, от первичного ключа, а в случае если этот ключ составной, то от каждого его элемента. Полной зависимостью называется возможность однозначно определять значения каждого неключевого поля при помощи значения первичного ключа. В качестве первичного ключа рекомендуется использовать самые простые данные, которые имеют уникальные значения. Поэтому для всех таблиц определяются уникальный идентификатор «id». Так для объекта «технические средства» определяется новый атрибут, который является первичным ключом таблицы. Изменения представлены в таблице 8.

Таблица 8 – Идентификация объекта «технические средства»

ТЕХНИЧЕСКОЕ СРЕДСТВО / TECHNICAL_MEANS	
СВОЙСТВО ОБЪЕКТА И ЕГО ОПИСАНИЕ	ИМЯ АТТРИБУТА
Первичный ключ таблицы	id_ts
Наименование технического средства	name
Модель технического средства	model
Серийный номер	serial_number
Инвентарный номер	inventory_number
Год выпуска	release_year
Дата ввода в эксплуатацию	start_exploitation
Параметр1	option1
Параметр2	option2
Параметр3	option3
Параметр4	option4
Параметр5	option5
Параметр6	option6
Параметр7	option7

Окончание таблицы 8

Параметр8	option8
Параметр9	option9
Статус эксплуатации	status

Подобным образом производится идентификация объектов «материально-ответственные лица», «автоматизированное рабочее место», «местоположение» «статус» «параметры технических средств» и «пользователи системы».

Для приведения таблиц к третьей нормальной форме необходимо обеспечить чтобы столбцы, не являющиеся ключевыми, зависели от первичного ключа таблицы и не зависели от всех других столбцов. Третья нормальная форма гарантирует возможность корректирования значения любого поля, не являющегося первичным ключом, без воздействия на данные других полей. Приведение третьей нормальной форме подразумевает предыдущее преобразование данных к первой и второй нормальной форме.

Данные полученные после нормализации графически удобно представить в виде модели сущность-связь нотации Баркера. Ранее описанная модель Чена получила дальнейшее развитие в работах Баркера, описавшего оригинальную нотацию, которая позволяет интегрировать предложенные Питером Ченом средства описания информационной модели. В нотации Баркера используются только диаграммы сущность-связь ERD.

На ERD сущность представляется прямоугольником любого размера, содержащим имя сущности и список имен атрибутов, а также указатели ключевых атрибутов и связи между объектами. Разработанная модель Баркера представлена в приложении Б.

Графически разграничение прав доступа пользователей можно представить на функциональной схеме акторов системы представленной на рисунке 4.

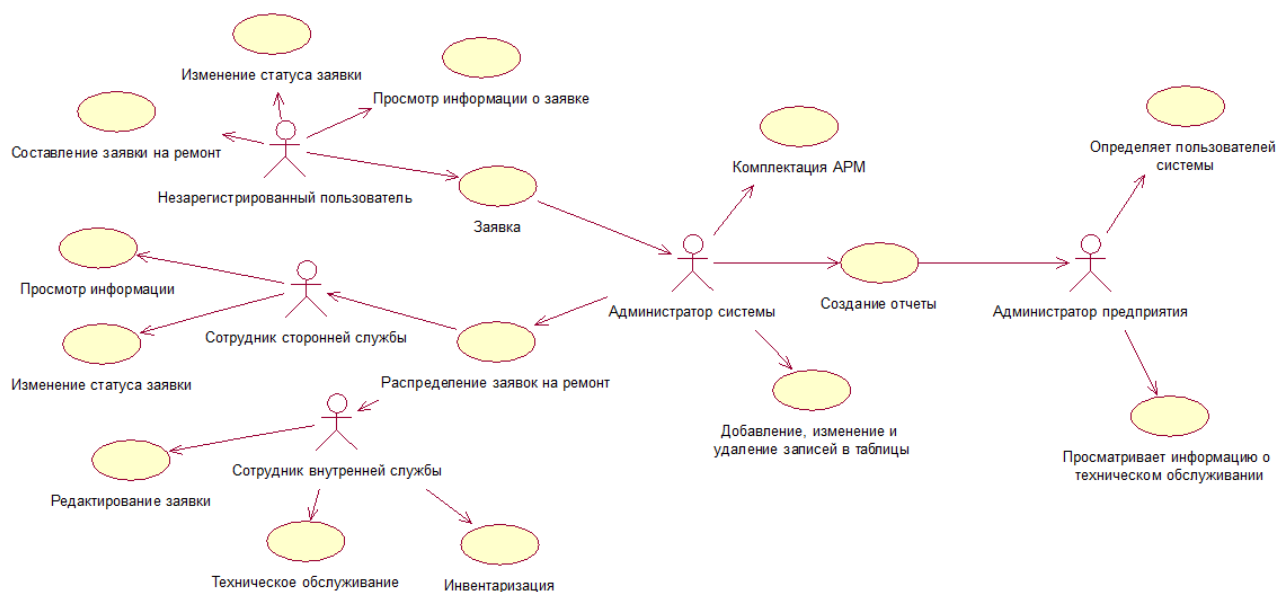


Рисунок 4 – Функциональная схема акторов системы

Незарегистрированные пользователи оставляют сообщение о поломке оборудования, составляя заявку на ремонт. После регистрации заявки они могут получить информацию о ней по предоставленному номеру.

Сотрудники внутренней службы имеют полный доступ к данным о техническом обслуживании и поступающим от пользователей заявкам.

Сотрудники сторонних служб имеют частичный доступ к информации о техническом обслуживании и заявкам пользователей.

Администратор системы имеет доступ ко всей информации из базы данных с возможностью добавления, изменения и удаления данных.

Администратор предприятия может просматривать основную информацию о техническом обслуживании и инвентаризации.

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

3 Реализация программного модуля

3.1 Реализация базы данных

Создание комплекса программ осуществляется при помощи портативной серверной платформы «Open Server Panel».

Для начала необходимо создать папку в каталоге «domains», в которой впоследствии будет находиться вся необходимая информация и код программ, организующих решение поставленных задач.

Для начала работы необходимо запустить данную сборку, нажав на исполняемый файл «Open Server.exe».

При запуске программной среды открывается меню с параметрами и настройками, содержащее в себе все необходимые компоненты для веб-разработки. Для реализации базы данных следует запустить приложение «RНРMyAdmin», находящееся на вкладке «Дополнительно». С его помощью легко создавать и администрировать базы данных.

Компонентная структура соответствует трехзвенной модели информационных систем, которая широко себя зарекомендовала и занимает лидирующие позиции при выборе архитектуры современных ИС. При разработке использованы стандартные звенья «Клиент» – «Сервер приложений – веб-сервер» – «СУБД».

Система управления базами данных MySQL имеет развитую систему доступа к базам данных. Пользователь базы данных может иметь доступ ко всей базе данных, отдельным таблицам и определённым столбцам таблиц. MySQL реализует ограничение действий производимых пользователем с записями. Для организации достаточно непростой структуры доступа используются таблицы в специальной базе данных. На основании значений этих таблиц выстраивается политика предоставления доступа.

База данных будет создаваться в веб-приложении для администрирования баз данных PHPMyAdmin. Данное приложение позволяет с помощью интерфейса создавать сущности базы данных с указанием необходимых атрибутов и их ограничений. При этом автоматически генерируются запросы к СУБД MySQL, которые среда дублирует на экране после выполнения, а также выводится сообщение о результате команды.

Во время создания базы данных выполняется команда CREATE DATABASE. При этом среда предлагает на выбор десятки кодировок. Наиболее распространенными на данный момент являются ASCII и Unicode. Так как ASCII содержит неполный набор алфавитов, наиболее приемлемо использовать кодировку Unicode. Тогда при вводе данных в форму символами любого алфавита они будут корректно распознаны и внесены в таблицу базы данных без изменений. Остановим выбор на кодировке UTF8-general-ci.

После создания базы данных необходимо заполнить её рядом таблиц, описанных при разработке базы данных. Для функционирования базы данных необходимо заполнить созданные таблицы в соответствии с моделью данных.

Реализовать все сущности, отраженные в инфологической модели, можно при помощи интерфейса веб-приложения. При создании таблицы предлагается указать название и количество ее атрибутов. После чего необходимо заполнить поля необходимыми данными. Создание таблицы и определение её атрибутов представлено на рисунке 5.

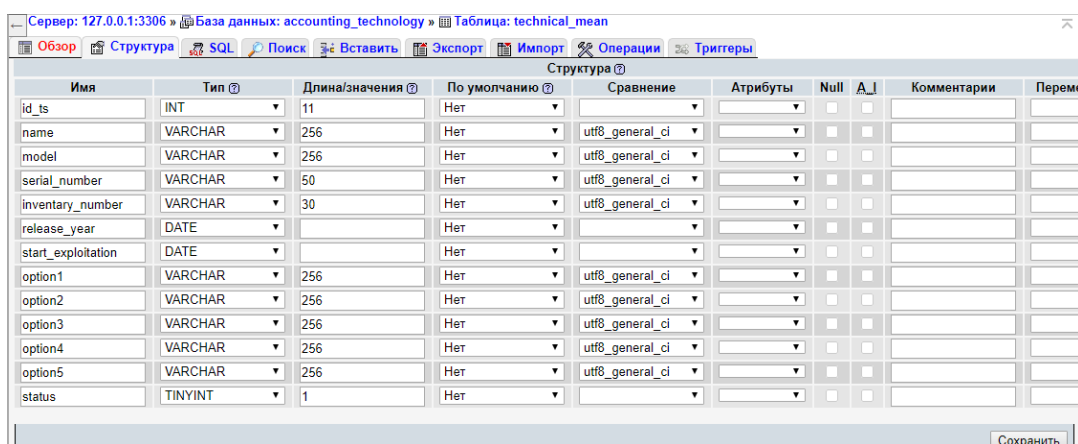


Рисунок 5 – Создание таблицы «технические средства»

3.2 Создание интерфейсной части модуля

При разработке любого программного изделия необходимо учитывать качество и понятность интерфейса. Качество интерфейса напрямую зависит от того, насколько понятно и правильно он будет разработан.

Немаловажным фактором при разработке интерфейса является наличие цветовой гаммы. Выбор цветов при разработке интерфейса необходим для того, чтобы пользователь смог воспринимать информацию должным образом. Поэтому выбрано минимальное количество цветов для проектирования программного изделия. Это классические белый, черный и спокойные оттенки серого цвета. Так как эти цвета не слишком броские, то пользователю будет удобно и приятно воспринимать информацию.

Однако при реализации интерфейса, необходимо учитывать создание кнопок меню и перехода, а также другие элементы, необходимые для реализации поставленных задач.

Для реализации учёта технических средств необходимо создать следующие элементы:

- а) главное меню, содержащее ссылки на информацию об организации, основные компоненты и приоритетную информацию для быстрого доступа;
- б) индивидуальное меню, доступное зарегистрированному пользователю;
- в) основное пространство, предназначенное для реализации функций.

Меню приложения будет располагаться в верхней части главной формы. Интерфейс форм будет содержать поля для ввода данных, а также кнопки для реализации необходимой функции. Также разработан интерфейс для оперирования информацией базы данных MySQL.

Для решения поставленных задач, необходимо создание нескольких страниц, имеющих соединение с базой данных. Первоначальной задачей


```

<ul>
<li><a href="index.php" class="button special"> Главная</a></li>
<li><a href="request.php">Подать заявку</a></li>
<li><a href="service.php">Сервис служб</a></li>
</ul>
</nav>
</header>

```

При создании решения задачи авторизации пользователей, учёта технических средств, ведения технического обслуживания, необходима реализация вводной компоненты задачи. Для этого на форму необходимо добавить элементы, в функционал которых входит считывание данных вводимых пользователем и отправка этих данных на сервер для последующего помещения информации в базу данных. Элемент «<form> </form>» представляет секцию документа, содержащую интерактивные элементы управления, которые позволяют пользователю отправлять информацию на веб-сервер. В качестве параметра определён метод запроса «post», являющийся методом передачи элементов формы. Для реализации интерактивности сайта используется тег «<input> </input>». Основным параметром данного тега является «type», определяющий вид интерактивного элемента. Например, «type="text"» предоставляет пользователю возможность ввода текста в поле на форме, а «type="submit"» является кнопкой для передачи введённых данных на сервер. Форма авторизации пользователя представлена в листинге 2.

Листинг 2 – Форма авторизации пользователя

```

<form method="post" action="auth.php">
<div>
<div class="бу 12u$(4)">
<h3>Авторизация</h3>
<input type="text" name="login" id="name" value="" placeholder="Логин"
/>
<input type="password" name="password" id="email" value=""
placeholder="Пароль" /></br>
</div>
<div class="12u$">
<ul class="actions">

```


3.3 Реализация процедур модуля

После разработки элементов интерфейса пользователя необходимо перейти к реализации функциональной части. Все программы комплекса находятся на локальном сервере в папке «domains».

«*index.php*» – основное меню задачи. Содержит основной интерфейс программного комплекса для инвентаризации компьютерной техники и программного обеспечения, описанный в файле «*style.css*», а также общую информацию о разрабатываемой системе государственного бюджетного учреждении здравоохранения «Поликлиника №4». Имеет меню для перехода на соответствующие страницы.

После нажатия на кнопку главного меню «Сервис служб» пользователь попадает на страницу авторизации, где требуется ввести логин и пароль, ранее определенный администратором системы. Страница авторизации описана в файле «*auth.php*». Кроме элементов интерфейса в данном файле описывается запрос к таблице базы данных, содержащую информацию о пользователях системы. При нажатии кнопки «submit» с соответствующим именем данные с формы записываются в переменные. Описание функций класса «Database» находится в файле «*config.php*». Путем SQL-запроса в базе данных определяется пользователь с указанным логином. Далее следует проверка соответствия введенного пользователем пароля и пароля из базы данных. Если данные совпадают, то информация о пользователе попадает на страницу «*service.php*». Код авторизации представлен в листинге 4.

Листинг 4 – Код авторизации пользователей

```
<?
session_start();
require_once 'config.php';
if (isset($_POST['submit'])) {
$password = $_POST['password'];
```


благодаря которым реализуется редирект на сервисные страницы. Проверка авторизации представлена в листинге 6.

Листинг 6 – Разграничение прав доступа

```
session_start();
if (!isset($_SESSION['login'])) {
header("Location: /auth.php");
}
require_once 'config.php';
$db = new Database();
$login = $_SESSION['login'];
getRow = $db->getRow("SELECT * FROM users WHERE login = ?",
["$login"]);
if ($getRow <> null) {
$_SESSION['flag'] = $getRow['flag'];
if ($getRow['flag'] === "a"){header("Location: /servicea.php");}
elseif ($getRow['flag'] === "b"){header("Location: /serviceb.php");}
elseif ($getRow['flag'] === "c"){header("Location: /servicec.php");}
elseif ($getRow['flag'] === "d"){header("Location: /serviced.php");}
$db->Disconnect();
}else{
unset($_SESSION['login']);
header("Location: /auth.php");
}
```

В случае если данные логина не были получены или была неудачно пройдена идентификация, пользователь попадёт на страницу авторизации, где повторно будут запрошены данные логина и пароля. Если же идентификация была пройдена успешно, пользователь попадет на соответствующую его правам доступа страницу. Данная страница содержит основную информацию о пользователе и меню с соответствующими разграничениям функциями. Сотрудник внутренней службы будет перенаправлен на страницу «servicea.php». Сотрудник сторонней службы попадет на страницу «serviceb.php». Администратор системы – «servicec.php». Администратор предприятия соответственно на «serviced.php».

Пользователи имеют доступ к определенным таблицам базы данных. Базовая функция в реализуемой системе – организация ввода и вывода

информации. «*add.php*» – страница, содержащая в себе вводную компоненту задачи учёта технических средств. Добавление данных организовано при помощи тега «*input*», имеющего соответствующий тип. В случае ввода текстовых и числовых значений, используется тип «*text*». Если необходимо ввести дату, используется тип «*date*». Все введённые в определённые поля данные отправляются на сервер в базу данных, хранящую информацию о технических средствах. Добавление организовано при помощи SQL-запроса «*INSERT INTO*». Для данного запроса указываются таблицы, в которые будут добавлены новые данные и вводимые пользователем значения, которые указываются в запросе после «*VALUE*». На страницу ввода данных можно перейти при нажатии кнопки «Добавить» из страницы, выводящей таблицу со всеми сведениями из таблицы с описанием технических средств. Вывод данных описан в файле «*table.php*».

Файл «*table.php*» содержит код, организующий вывод данных из таблицы базы данных в таблицу, находящуюся на форме. Вывод данных производится для каждой таблицы, выбранной пользователем. В зависимости от выбранной таблицы устанавливается определенный параметр, который можно увидеть в адресной строке после символа «*?*». Например, «*table.php?table=technical_mean*» реализует вывод информации из таблицы базы данных «*technical_mean*». Для организации поиска данных и вывода информации на печать используется скриптовый язык программирования JavaScript, помещенный между тегами «*script*» и «*/script*». Реализация функции вывода информации на печать представлена в листинге 7.

Листинг 7 – Функция печати

```
<script type="text/javascript">
function printData()
{
    var divToPrint=document.getElementById("printTable");
    newWin= window.open("");
    newWin.document.write(divToPrint.outerHTML);
    newWin.print();
}
```


3.4 Тестирование программного модуля

Перед передачей разработанного проекта заказчику и его запуском в широкое пользование, необходимо обязательно проверить программу на предмет ошибок и недоработок. Другими словами провести тестирование.

Тестирование необходимо для выявления багов, которые в последствии могут нанести большой урон системе, что в свою очередь может привести к непредвиденному материальному ущербу. Во избежание многочисленных проблем лучше своевременно найти и исправить ошибки перед запуском системы в интернет.

Проверка новой разработки задача не только тестировщиков, но в первую очередь всей команды, которая работала над созданием. И пренебрегать данным этапом крайне нежелательно.

Для начала – проверка всех текстов на соответствие правилам языка. Любой текст, выложенный на сайте, должен соответствовать орфографическим, пунктуационным, грамматическим и лексическим правилам используемого языка. Это касается всего текстового контента – не только больших статей, но и коротких заголовков страниц, надписей на кнопках и прочего.

При проверке были выявлены некоторые недочёты. Система не до конца русифицирована, что в свою очередь может усложнить эксплуатацию.

Следующий этап – визуальная оценка разработанной системы. Каждая картинка на сайте должна быть оптимизирована. Во избежание сложностей при загрузке страниц проверяется размер и адаптированность изображений. Также следует проверить подписи ко всем фотографиям, их описания, заголовки, альтернативный текст.

На основании проведённой проверки графических объектов ошибок обнаружено не было.

Предполагается, что формы будут заполнять простые пользователи, поэтому в процесс заполнения должен быть предельно понятным. На этом этапе можно произвести тестирование на критические данные.

Проверка на критические данные отражена в таблице 9.

Таблица 9 – Критическое тестирование

Наименование	Описание тестового случая	Результат
Главная форма	Нажатие на кнопку «Главная».	Кнопка нажата. Переход по ссылке произошёл.
	Нажатие на кнопку «Подать заявку».	Кнопка нажата. Переход по ссылке произошёл.
	Нажатие на кнопку «Сервис служб».	Кнопка нажата. Переход по ссылке произошёл.
Форма авторизации	Ввод корректного логина и корректного пароля (логин «Камаева», пароль «1234»). Нажатие на кнопку «Войти».	Кнопка нажата. Произошёл переход в личный кабинет администратора системы. Авторизация прошла успешно.
	Ввод корректного логина и некорректного пароля (логин «Камаева», пароль «1»). Нажатие на кнопку «Войти».	Кнопка нажата. Произошла переадресация на страницу авторизации. Требуется повторная авторизация.
	Ввод некорректного логина и корректного пароля (логин «Камаеваааа», пароль «1234»). Нажатие на кнопку «Войти».	Кнопка нажата. Произошла переадресация на страницу авторизации. Требуется повторная авторизация.

Продолжение таблицы 9

	Ввод некорректного логина и некорректного пароля (логин «Камаевааа», пароль «1»). Нажатие на кнопку «Войти».	Кнопка нажата. Произошла переадресация на страницу авторизации. Требуется повторная авторизация.
	Ввод только корректного логина (логин «Камаева»). Нажатие на кнопку «Войти».	Кнопка нажата. Произошла переадресация на страницу авторизации. Требуется повторная авторизация.
	Ввод только корректного пароля (пароль «1234»). Нажатие на кнопку «Войти».	Кнопка нажата. Произошла переадресация на страницу авторизации. Требуется повторная авторизация.
	Нажатие на кнопку «Войти».	Кнопка нажата. Произошла переадресация на страницу авторизации. Требуется повторная авторизация.
	Ввод корректного логина и корректного пароля (логин «Камаева», пароль «1234»). Нажатие на кнопку «Отмена».	Кнопка нажата. Введённые данные из текстовых полей исчезли.
Работа с табличными данными	Выбор таблицы из выпадающего списка (таблица «Материально-ответственные лица»). Нажатие на кнопку	Кнопка нажата. Произошёл переход на страницу с параметром выбранной таблицы (таблица responsible_for_commissioning).

Продолжение таблицы 9

	«Ввод данных».	Организован вывод информации из вызываемой таблицы. На форме находятся кнопки, реализующие функции работы с табличными данными.
Добавление данных в таблицу	На форме с табличными данными нажатие кнопки «Добавить».	Переход на форму вводной компоненты задачи, содержащей поля для ввода и кнопки «Добавить» и «Отмена».
	Ввод корректных значений (ФИО «Камаева Анастасия Андреевна», профессия «программист», АРМ «кабинет 201»). Нажатие кнопки «Добавить».	Успешное добавление новой записи. Переход на страницу выводящую обновлённую таблицу.
	Ввод пустых значений (ФИО «NULL», профессия «NULL», АРМ «NULL»). Нажатие кнопки «Добавить».	Успешное добавление новой записи, содержащую значения «NULL». Переход на страницу выводящую обновлённую таблицу.
Поиск по таблице	Ввод одного символа («а»).	Автоматическое обновление таблицы в соответствии с поисковым запросом. Вывод записей, содержащих «а».

Окончание таблицы 9

	Удаление символа «а» из поискового запроса.	Автоматическое обновление таблицы в соответствии с поисковым запросом. Вывод всех записей таблицы.
Изменение данных таблицы	На форме с табличными данными нажатие ссылки «Изменить».	Переход на форму вводной компоненты задачи, содержащей заполненные поля для ввода и кнопки «Изменить» и «Отмена».
	Изменение данных корректными значениями (профессия «инженер-программист»). Нажатие кнопки «Изменить».	Успешное изменение записи таблицы. Переход на страницу выводящую обновлённую таблицу.
	Удаление информации, то есть передача пустого значения (ФИО «NULL»). Нажатие кнопки «Изменить».	Успешное добавление новой записи, содержащую значения «NULL». Переход на страницу выводящую обновлённую таблицу.
Удаление записи таблицы	На форме с табличными данными нажатие ссылки «Удалить».	Успешное удаление записи таблицы. Перезагрузка страницы с обновлённой таблицей.
Вывод таблицы на печать	На форме с табличными данными нажатие кнопки «Печать».	Успешный переход на страницу настройки печати.

Изм.	Лист	№ докум.	Подпись	Дата

ДП-02069964-090203-07-18

Лист

50

ЗАКЛЮЧЕНИЕ

В ходе выполнения дипломного проекта был произведён анализ и составление технической документации, описание предметной области, в соответствии с техническими требованиями, закреплены навыки по работе с СУБД и созданием программного модуля для более удобной работы с базой данных.

В результате предпроектного обследования на основе сведений, предоставленных организацией было принято решение реализации модуля на основе веб-технологий. Для хранения информации соответственно выбрана база данных MySQL. Для управления этими базами данных – веб-приложение PHPMyAdmin, которое располагается на сетевых ресурсах Open Server.

На основе изучения предметной области реализовано проектирование и реализация базы данных, интерфейса пользователя и модуля технического обслуживания и инвентаризации.

В данном дипломном проекте разработана база данных для ГБУЗ РМ «Поликлиника №4» и создан программный модуль учёта технических средств. Назначение разработки заключается в следующем: обеспечить удобную работу сотрудников предприятия и повысить производительность.

В результате была разработана информационная система для обслуживания технических средств и оборудования.

В итоге, достигнута цель и реализованы задачи, обозначенные в начале написания дипломного проекта. В системе выполняются все необходимые требования.

Система имеет возможность расширения. Возможно добавление новых функций за счет доработки кода серверных сценариев и оптимизация под мобильные устройства.

					<i>ДП-02069964-090203-07-18</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		52

Режим доступа: http://www.consultant.ru/document/cons_doc_LAW_34683/36f9a2e4f68547adf3ced702023adb570d777ea8/.

19 Цикритизис Д. Модели данных / Д. Цикритизис, Ф. Лоховски – М. : Финансы и статистика, 1985. – 344 с.

					<i>ДП-02069964-090203-07-18</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		55

Продолжение ПРИЛОЖЕНИЯ А

3. ХАРАКТЕРИСТИКА ОБЪЕКТОВ АВТОМАТИЗАЦИИ

Объектами автоматизации являются процессы, возникающие в ходе оказания услуг, связанных с предоставлением информации о технических средствах пользователям.

3.1. Описание функциональных систем

В состав автоматизированной информационной системы «Учёт технических средств» должны входить следующие подсистемы:

1. Учёт технических средств

Предусматривает возможности:

- ввод;
- редактирование;
- поиск;
- списание.

Отчёты:

- количество технических средств определённого класса;
- списание технических средств;
- поиск по полям(свойствам).

2. Планирование технического обслуживания технических средств

Предусматривает возможности:

- создание плана;
- редактирование плана;
- удаление плана.

Отчёты:

- сам план;
- генерация плана;
- контроль над исполнением плана.

3. Учёт пользователей технических средств и материально ответственных лиц

Предусматривает возможности:

- ввод;
- редактирование;
- поиск.

Отчёты:

- количество пользователей определённого технического средства

4. Журнал технического обслуживания для технических средств

Предусматривает возможности:

- добавление событий;
- редактирование событий.

Отчёты:

- ремонтные работы на даты;
- все ремонтные работы, проводимые с техническим средством;
- технические работы, проводимые в период.

6. Автоматизированное рабочее место пользователей

Предусматривает возможности:

– создание автоматизированного рабочего места из технических средств для пользователей.

7. Аутентификация пользователей

					<i>ДП-02069964-090203-07-18</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		57

ПРИЛОЖЕНИЕ Б (обязательное)

Диаграммы базы данных

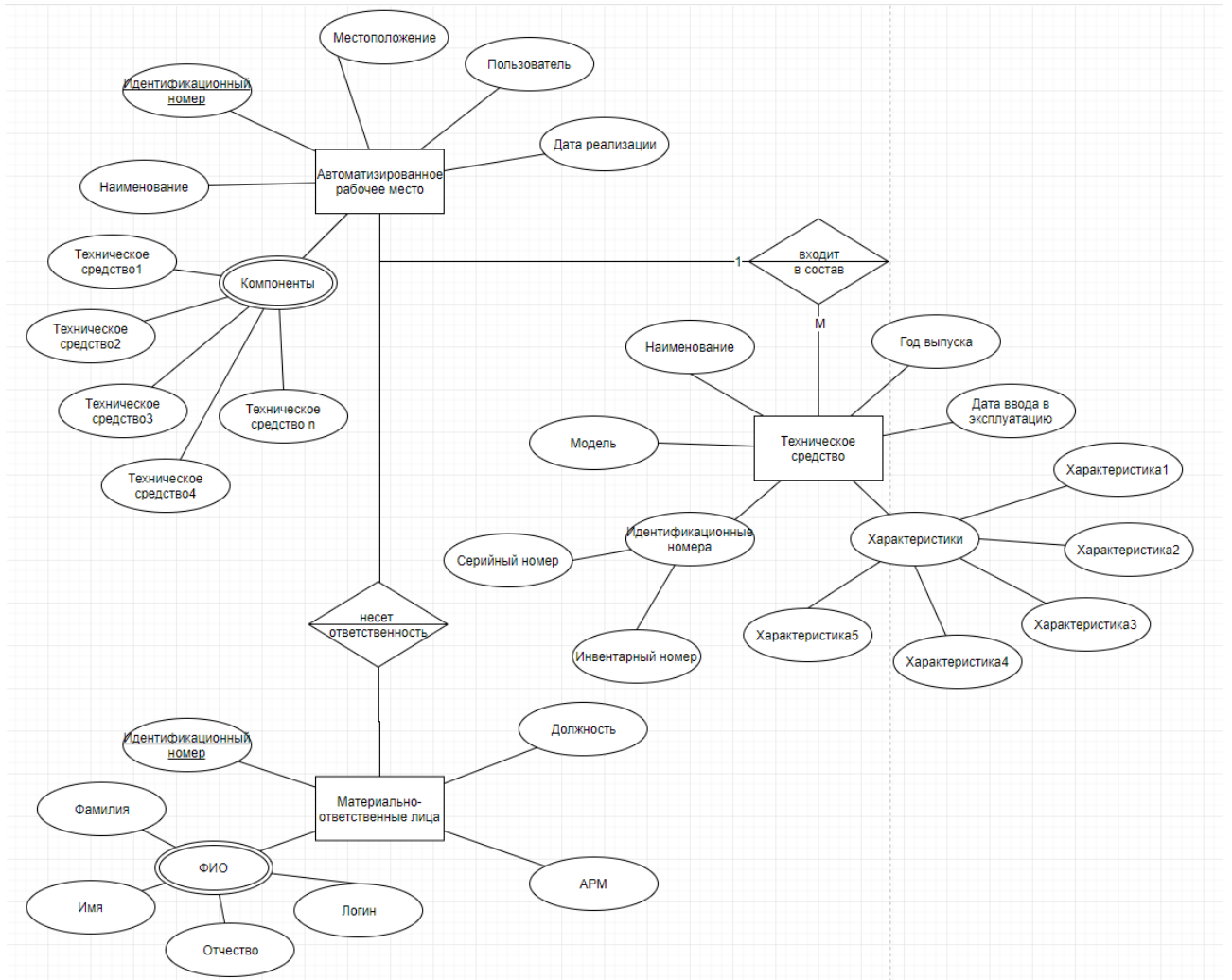


Рисунок Б.1 – ER-диаграмма базы данных

Окончание ПРИЛОЖЕНИЯ Б

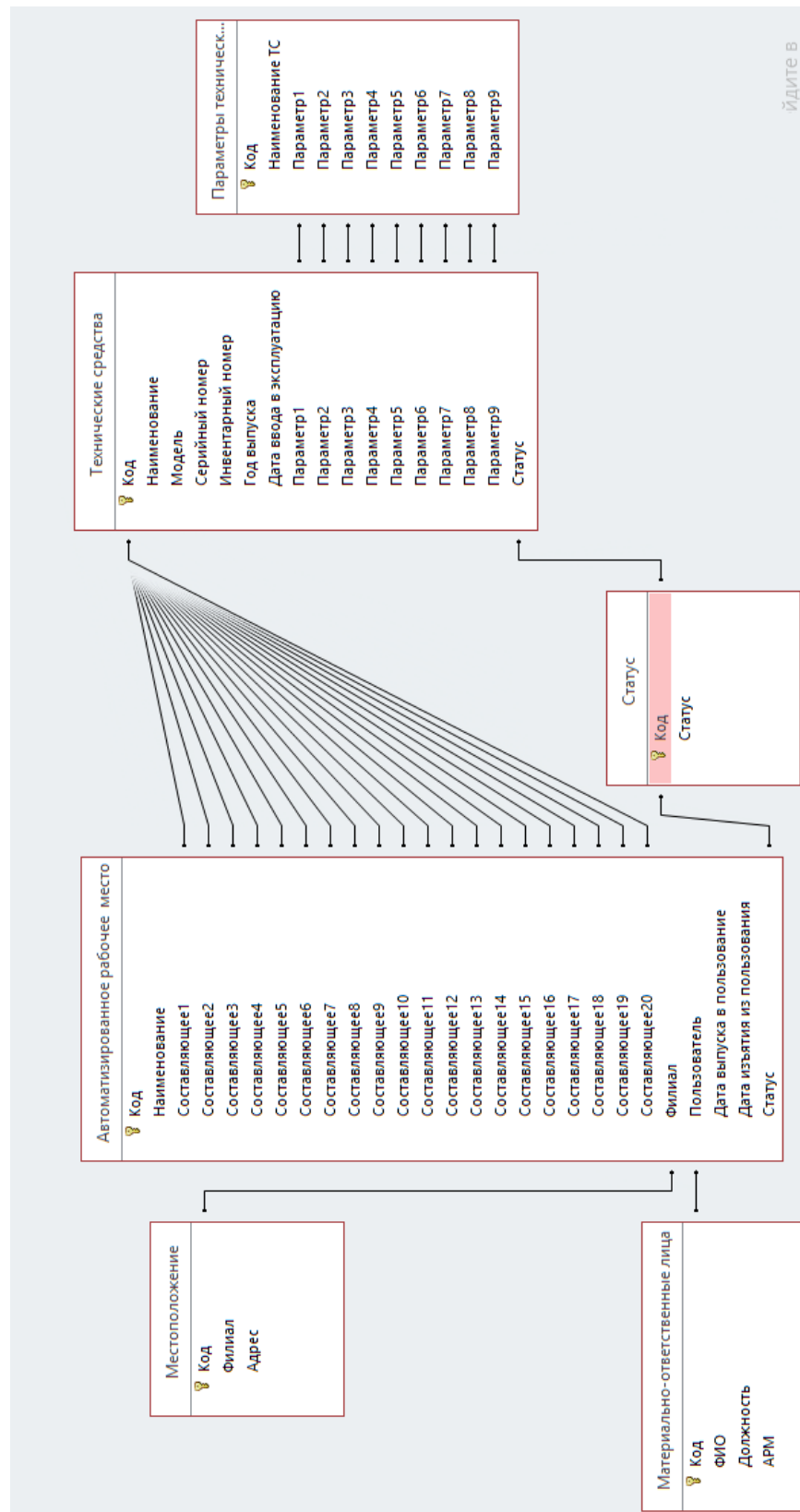


Рисунок Б.2 – Схема данных

ПРИЛОЖЕНИЕ В

(обязательное)

Листинг функций программного модуля

1 Файл «index.php» – главная страница

```
<!DOCTYPE html>
<html lang="ru">
  <head>
    <meta charset="UTF-8">
    <title>Главная</title>
    <meta http-equiv="content-type" content="text/html; charset=utf-8" />
    <!--[if lte IE 8]><script src="js/html5shiv.js"></script><![endif]-->
    <script src="js/jquery.min.js"></script>
    <script src="js/skel.min.js"></script>
    <script src="js/skel-layers.min.js"></script>
    <script src="js/init.js"></script>
    <noscript>
      <link rel="stylesheet" href="css/skel.css" />
      <link rel="stylesheet" href="css/style.css" />
      <link rel="stylesheet" href="css/style-xlarge.css" />
    </noscript>
  </head>
  <body class="landing">

    <!-- Header -->
    <header id="header">
      <nav id="nav">
        <ul>
          <li><a href="index.php" class="button special">Главная</a></li>
          <li><a href="request.php">Подать заявку</a></li>
          <li><a href="service.php">Сервис служб</a></li>
        </ul>
      </nav>
    </header>

    <!-- Banner -->
    <section id="banner">
      <h2>Информационная система "Обслуживание технических средств и
оборудования"</h2><br>
      <p>Содержит модули:</p>
      <i>Журнал технического обслуживания и инвентаризация</i><br>
      <i>Обработка заявок на ремонт</i><br>
      <i>Планирование технического обслуживания</i><br>
      <i>на основе документации ГБУЗ РМ "Поликлиника №4"</i>
    </section>

    <!-- One -->
    <section id="one" class="wrapper style1 special">
      <div class="container">
        <header class="major">
          <h2>FAQ</h2>
        </header>
        <div class="row 150%">
          <div class="4u 12u$(medium)">
```

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

ДП-02069964-090203-07-18

Лист

61

Продолжение ПРИЛОЖЕНИЯ В

```
<script src="js/skel-layers.min.js"></script>
<script src="js/init.js"></script>
<noscript>
  <link rel="stylesheet" href="css/skel.css" />
  <link rel="stylesheet" href="css/style.css" />
  <link rel="stylesheet" href="css/style-xlarge.css" />
</noscript>
</head>
<body>

<!-- Header -->
  <header id="header">
    <nav id="nav">
      <ul>
        <li><a href="index.php">Главная</a></li>
        <li><a href="request.php">Подать заявку</a></li>
        <li><a href="service.php" class="button special">Сервис служб</a></li>
      </ul>
    </nav>
  </header>

<!-- One -->
<center>
  <section id="main" class="wrapper">
    <div class="container">
      <form method="post" action="auth.php">
        <div>
          <div class="6u 12u$(4)">
            <h3>Авторизация</h3>
            <input type="text" name="login" id="name" value=""
placeholder="Логин" />
            <input type="password" name="password" id="email"
value="" placeholder="Пароль" /></br>
            </div>
            <div class="12u$">
              <ul class="actions">
                <li><input type="submit" name="submit"
value="Войти" class="special" /></li>
                <li><input type="reset" value="Отмена" /></li>
              </ul>
            </div>
          </div>
        </div>
      </form>
    </div>
  </section></center><br />
</body>
</html>
```

3 Файл «config.php» – реализация доступа к базе данных

```
<?php
$th4 = ["id_workstation" => "ID", "Наименование" => "Наименование", "Компонент" =>
"Компонент", "Компонент" => "Компонент", "Компонент" => "Компонент", "Компонент" =>
"Компонент", "Компонент" => "Компонент", "Компонент" => "Компонент", "Компонент" =>
"Компонент", "Компонент" => "Компонент", "Компонент" => "Компонент"];
```

Изм.	Лист	№ докум.	Подпись	Дата

ДП-02069964-090203-07-18

Лист

63

Продолжение ПРИЛОЖЕНИЯ В

```
$td4 = ["id_workstation" => "id_workstation", "workstation" => "workstation", "component1" =>
"component1", "component2" => "component2", "component3" => "component3", "component4" =>
"component4", "component5" => "component5", "component6" => "component6", "component7" =>
"component7", "component7" => "component7", "component8" => "component8", "component9" => "component9"];
$th1 = ["id" => "ID", "Логин" => "Логин", "Пароль" => "Пароль", "Роль" => "Роль", "Права доступа" => "Права
доступа"];
$td1 = ["id" => "id", "login" => "login", "password" => "password", "position" => "position", "flag" => "flag"];
$th2 = ["id_ts" => "ID", "Наименование" => "Наименование", "Модель" => "Модель", "Серийный номер" =>
"Серийный номер", "Инвентарный номер" => "Инвентарный номер", "Год выпуска" => "Год выпуска", "Дата
ввода в эксплуатацию" => "Дата ввода в эксплуатацию", "Параметр" => "Параметр", "Параметр" =>
"Параметр", "Параметр" => "Параметр", "Параметр" => "Параметр", "Параметр" => "Параметр", "Параметр" =>
"Параметр", "Параметр" => "Параметр", "Параметр" => "Параметр", "Параметр" => "Параметр", "Статус" =>
"Статус"];
$td2 = ["id_ts" => "id_ts", "name" => "name", "model" => "model", "serial_number" =>
"serial_number", "inventory_number" => "inventory_number", "release_year" => "release_year", "start_exploitation" =>
"start_exploitation", "option1" => "option1", "option2" => "option2", "option3" => "option3", "option4" =>
"option4", "option5" => "option5", "option6" => "option6", "option7" => "option7", "option8" => "option8", "option9" =>
"option9", "status" => "status"];
$th3 = ["id" => "ID", "Технические Средства" => "Технические Средства", "Дата" => "Дата", "Статус" =>
"Статус"];
$td3 = ["id_plan" => "id_plan", "technical_means" => "technical_means", "date_plan" => "date_plan", "status" =>
"status"];
$th5 = ["id" => "ID", "Наименование" => "Наименование", "Параметр" => "Параметр", "Параметр" =>
"Параметр", "Параметр" => "Параметр", "Параметр" => "Параметр", "Параметр" => "Параметр", "Параметр" =>
"Параметр", "Параметр" => "Параметр", "Параметр" => "Параметр"];
$td5 = ["id" => "id", "name" => "name", "option1" => "option1", "option2" => "option2", "option3" =>
"option3", "option4" => "option4", "option5" => "option5", "option6" => "option6", "option7" => "option7", "option8" =>
"option8", "option9" => "option9"];
$th6 = ["id" => "ID", "Статус" => "Статус"];
$td6 = ["id_status" => "id_status", "status" => "status"];
$th7 = ["id" => "ID", "ФИО" => "ФИО", "Профессия" => "Профессия", "АРМ" => "АРМ"];
$td7 = ["id_person" => "id_person", "full_name" => "full_name", "profession" => "profession", "job" => "job"];
$th8 = ["id" => "ID", "Филиал" => "Филиал", "Адрес" => "Адрес"];
$td8 = ["id" => "id", "branch" => "branch", "address" => "address"];
class Database{
    public $isConn;
    protected $datab;

    // connect to db
    public function __construct($username = "Frizzes", $password = "Frizzes14", $host = "localhost", $dbname =
"accounting_", $options = []){
        $this->isConn = TRUE;
        try {
            $this->datab = new PDO("mysql:host={$host};dbname={$dbname};charset=utf8", $username, $password,
$options);
            $this->datab->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
            $this->datab->setAttribute(PDO::ATTR_DEFAULT_FETCH_MODE, PDO::FETCH_ASSOC);
        } catch (PDOException $e) {
            throw new Exception($e->getMessage());
        }
    }

    public function Disconnect(){
        $this->datab = NULL;
        $this->isConn = FALSE;
    }
}
```

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

ДП-02069964-090203-07-18

Лист

64

Продолжение ПРИЛОЖЕНИЯ В

```
// get row
public function getRow($query, $params = []){
    try {
        $stmt = $this->datab->prepare($query);
        $stmt->execute($params);
        return $stmt->fetch();
    } catch (PDOException $e) {
        throw new Exception($e->getMessage());
    }
}

// get rows
public function getRows($query, $params = []){
    try {
        $stmt = $this->datab->prepare($query);
        $stmt->execute($params);
        return $stmt->fetchAll();
    } catch (PDOException $e) {
        throw new Exception($e->getMessage());
    }
}

// insert row
public function insertRow($query, $params = []){
    try {
        $stmt = $this->datab->prepare($query);
        $stmt->execute($params);
        return TRUE;
    } catch (PDOException $e) {
        throw new Exception($e->getMessage());
    }
}

// update row
public function updateRow($query, $params = []){
    $this->insertRow($query, $params);
}

// delete row
public function deleteRow($query, $params = []){
    $this->insertRow($query, $params);
}
}
?>
```

4 Файл «servise.php» – распределение прав доступа пользователей в соответствии с флагом

```
<?
session_start();
if (!isset($_SESSION['login'])) {
    header("Location: /auth.php");
}
require_once 'config.php';
$db = new Database();
$login = $_SESSION['login'];
$getRow = $db->getRow("SELECT * FROM users WHERE login = ?", ["$login"]);
if ($getRow <> null) {
    $_SESSION['flag'] = $getRow['flag'];
}
```

										Лист
										65
Изм.	Лист	№ докум.	Подпись	Дата	ДП-02069964-090203-07-18					

Продолжение ПРИЛОЖЕНИЯ В

```
if($getRow['flag'] === "a"){header("Location: /servicea.php");}
elseif($getRow['flag'] === "b"){header("Location: /serviceb.php");}
elseif($getRow['flag'] === "c"){header("Location: /servicec.php");}
elseif($getRow['flag'] === "d"){header("Location: /serviced.php");}
    $db->Disconnect();
}
else{
    unset($_SESSION['login']);
    header("Location: /auth.php");
}
?>
```

5 Файл «servicea.php» – страница сотрудника внутренней службы

```
<?
session_start();
if($_SESSION['flag']<>'a' && $_SESSION['login'] != null){
header("Location: /service.php");
}
$_SESSION['true'] = $_SESSION['login'];
?>
<!DOCTYPE html>
<html lang="ru">
    <head>
        <meta charset="UTF-8">
        <title>Личный кабинет</title>
        <meta http-equiv="content-type" content="text/html; charset=utf-8" />
        <meta name="description" content="" />
        <meta name="keywords" content="" />
        <script src="js/jquery.min.js"></script>
        <script src="js/skel.min.js"></script>
        <script src="js/skel-layers.min.js"></script>
        <script src="js/init.js"></script>
        <noscript>
            <link rel="stylesheet" href="css/skel.css" />
            <link rel="stylesheet" href="css/style.css" />
            <link rel="stylesheet" href="css/style-xlarge.css" />
        </noscript>
    </head>
    <body>
        <!-- Header -->
        <header id="header">
            <nav id="nav">
                <ul>
                    <li><a href="index.php">Главная</a></li>
                    <li><a href="table.php?table=plan_maintenance">План
ТО</a></li>
                    <li><a href="table.php?table=technical_mean">Заполнение
справочников</a></li>
                    <li><a href="service.php" class="button special">Личный
кабинет</a></li>
                </ul>
            </nav>
        </header>
        <!-- One -->
        <section id="main" class="wrapper">
```

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

ДП-02069964-090203-07-18

Лист

66

Продолжение ПРИЛОЖЕНИЯ В

```
<div class="container">
  <section>
    <h4>Сотрудник внутренней службы</h4>
    <p><span class="image left"><?php echo ($_SESSION['login']);?></span>
  <section>
    <form action="update.php" method="post">
      <div class="table-wrapper">
        <table>
          <thead>
            <tr>
              <th>Логин</th>
              <th>Пароль</th>
              <th>Подтверждение пароля</th>
            </tr>
          </thead>
          <tbody>
            <tr>
              <td><input type="text" name="login" id="login" value=""
placeholder="Логин" /></td>
              <td><input type="password" name="password" id="password"
value="" placeholder="Пароль" /></td>
              <td><input type="password" name="retry" id="retry" value=""
placeholder="Подтверждение пароля" /></td>
            </tr>
          </tbody>
          <tfoot>
            <tr>
              <td colspan="2"><input type="submit" name="submit" value="Изменить"
class="special" /></td>
            </tr>
          </tfoot>
        </table>
      </form>
    </section>
  </div>
</p>
</section>
</div>
</body>
</html>
```

6 Файл «serviseb.php» – страница сотрудника сторонней службы

```
<?
session_start();
if ($_SESSION['flag']<>'b' && $_SESSION['login'] != null){
header("Location: /service.php");
}
$_SESSION['true'] = $_SESSION['login'];
?>
<!DOCTYPE html>
<html lang="ru">
  <head>
```

Продолжение ПРИЛОЖЕНИЯ В

```
<meta charset="UTF-8">
<title>Личный кабинет</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<meta name="description" content="" />
<meta name="keywords" content="" />
<script src="js/jquery.min.js"></script>
<script src="js/skel.min.js"></script>
<script src="js/skel-layers.min.js"></script>
<script src="js/init.js"></script>
<noscript>
  <link rel="stylesheet" href="css/skel.css" />
  <link rel="stylesheet" href="css/style.css" />
  <link rel="stylesheet" href="css/style-xlarge.css" />
</noscript>
</head>
<body>
<!-- Header -->
  <header id="header">
    <nav id="nav">
      <ul>
        <li><a href="index.php">Главная</a></li>
        <li><a href="table.php?table=plan_maintenance">План ТО</a></li>
        <li><a href="service.php" class="button special">Личный
кабинет</a></li>
      </ul>
    </nav>
  </header>
<!-- One -->
<section id="main" class="wrapper">
  <div class="container">
    <section>
      <h4>Сотрудник сторонней службы</h4>
      <p><span class="image left"><?php echo ($_SESSION['login']);?></span>
    <section>
      <form action="update.php" method="post">
        <div class="table-wrapper">
          <table>
            <thead>
              <tr>
                <th>Логин</th>
                <th>Пароль</th>
                <th>Подтверждение пароля</th>
              </tr>
            </thead>
            <tbody>
              <tr>
                <td><input type="text" name="login" id="login" value=""
placeholder="Логин" /></td>
                <td><input type="password" name="password" id="password" value=""
placeholder="Пароль" /></td>
                <td><input type="password" name="retry" id="retry" value=""
placeholder="Подтверждение пароля" /></td>
              </tr>
            </tbody>
          </table>
        </div>
      </form>
    </section>
  </div>
</section>
</body>
</html>
```

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

ДП-02069964-090203-07-18

Лист

68

Продолжение ПРИЛОЖЕНИЯ В

```
 </td> class="special" /></td> </tr> </tfoot> </table> </form> </section> </div> </p> </section> </div> </section> </body> </html> | |
```

7 Файл «service.php» – страница системного администратора

```

<?
session_start();
if ($_SESSION['flag']<>'c' && $_SESSION['login'] != null){
header("Location: /service.php");
}
$_SESSION['true'] = $_SESSION['login'];
?>
<!DOCTYPE html>
<html lang="ru">
  <head>
    <meta charset="UTF-8">
    <title>Личный кабинет</title>
    <meta http-equiv="content-type" content="text/html; charset=utf-8" />
    <meta name="description" content="" />
    <meta name="keywords" content="" />
    <script src="js/jquery.min.js"></script>
    <script src="js/skel.min.js"></script>
    <script src="js/skel-layers.min.js"></script>
    <script src="js/init.js"></script>
    <noscript>
      <link rel="stylesheet" href="css/skel.css" />
      <link rel="stylesheet" href="css/style.css" />
      <link rel="stylesheet" href="css/style-xlarge.css" />
    </noscript>
  </head>
  <body>
    <!-- Header -->
    <header id="header">
      <nav id="nav">
        <ul>
          <li><a href="index.php">Главная</a></li>
          <li><a href="table.php?table=technical_mean">Заполнение
справочников</a></li>
          <li><a href="table.php?table=users">Ввод
пользователей</a></li>
          <li><a href="service.php" class="button special">Личный
кабинет</a></li>
```

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

ДП-02069964-090203-07-18

Лист

69

Продолжение ПРИЛОЖЕНИЯ В

```

                                </ul>
                            </nav>
                        </header>
                    <!-- One -->
<section id="main" class="wrapper">
    <div class="container">
        <section>
            <h4>Системный администратор</h4>
            <p><span class="image left"><?php echo ($_SESSION['login']);?></span>
        <section>
            <form action="update.php" method="post">
                <div class="table-wrapper">
                    <table>
                        <thead>
                            <tr>
                                <th>Логин</th>
                                <th>Пароль</th>
                                <th>Подтверждение пароля</th>
                            </tr>
                        </thead>
                        <tbody>
                            <tr>
                                <td><input type="text" name="login" id="login" value=""
placeholder="Логин" /></td>
                                <td><input type="password" name="password" id="password" value=""
placeholder="Пароль" /></td>
                                <td><input type="password" name="retry" id="retry" value=""
placeholder="Подтверждение пароля" /></td>
                            </tr>
                        </tbody>
                    </table>
                    <tfoot>
                            <tr>
                                <td colspan="2"></td>
                                <td><input type="submit" name="submit" value="Изменить"
class="special" /></td>
                            </tr>
                        </tfoot>
                    </table>
                </form>
            </section>
        </div>
    </p>
</section>
</div>
</body>
</html>
```

8 Файл «servised.php» – страница администратора предприятия

```
<?
session_start();
if ($_SESSION['flag']<>'d' && $_SESSION['login'] != null){
header("Location: /service.php");
}
```

Продолжение ПРИЛОЖЕНИЯ В

```
$_SESSION['true'] = $_SESSION['login'];
?>
<!DOCTYPE html>
<html lang="ru">
  <head>
    <meta charset="UTF-8">
    <title>Личный кабинет</title>
    <meta http-equiv="content-type" content="text/html; charset=utf-8" />
    <meta name="description" content="" />
    <meta name="keywords" content="" />
    <script src="js/jquery.min.js"></script>
    <script src="js/skel.min.js"></script>
    <script src="js/skel-layers.min.js"></script>
    <script src="js/init.js"></script>
    <noscript>
      <link rel="stylesheet" href="css/skel.css" />
      <link rel="stylesheet" href="css/style.css" />
      <link rel="stylesheet" href="css/style-xlarge.css" />
    </noscript>
  </head>
  <body>
    <!-- Header -->
    <header id="header">
      <nav id="nav">
        <ul>
          <li><a href="index.php">Главная</a></li>
          <li><a href="table.php?table=none">Ввод данных</a></li>
          <li><a href="table.php?table=users">Пользователи</a></li>
          <li><a href="service.php" class="button special">Личный
кабинет</a></li>
        </ul>
      </nav>
    </header>
    <!-- One -->
    <section id="main" class="wrapper">
      <div class="container">
        <section>
          <h4>Администратор предприятия</h4>
          <p><span class="image left"><?php echo ($_SESSION['login']);?></span>
        <section>
          <form action="update.php" method="post">
            <div class="table-wrapper">
              <table>
                <thead>
                  <tr>
                    <th>Логин</th>
                    <th>Пароль</th>
                    <th>Подтверждение пароля</th>
                  </tr>
                </thead>
                <tbody>
                  <tr>
                    <td><input type="text" name="login" id="login" value=""
placeholder="Логин" /></td>
                    <td><input type="password" name="password" id="password"
value="" placeholder="Пароль" /></td>

```

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

ДП-02069964-090203-07-18

Лист

71

Продолжение ПРИЛОЖЕНИЯ В

```
placeholder="Подтверждение пароля" /></td>
</tr>
</tbody>
</tfoot>
</tr>
<td colspan="2"></td>
<td><input type="submit" name="submit" value="Изменить"
class="special" /></td>
</tr>
</tfoot>
</table>
</form>
</section>
</div>
</p>
</section>
</div>
</section>
</body>
</html>
```

9 Файл «tabl.php» – ВЫВОД ТАБЛИЧНЫХ ДАННЫХ

```
<?
session_start();
ini_set('display_errors', 0);
require_once 'config.php';
if (!isset($_SESSION['login'])) {
header("Location: /auth.php");
}
$flag = $_SESSION['flag'];
$table = $_GET['table'];
$_SESSION['table'] = $table;
if ($table <> "none") {
$db = new Database();
$rows = $db->getRows("SELECT * FROM $table");
} else {
if (isset($_POST['tablesubmit'])) {
$table = $_POST['tablnone'];
header("Location: /table.php?table=$table");
}
if ($table === "users") {$stds=$td1;$ths=$th1;}
elseif ($table === "technical_mean") {$stds=$td2;$ths=$th2;}
elseif ($table === "plan_maintenance") {$stds=$td3;$ths=$th3;}
elseif ($table === "workstation") {$stds=$td4;$ths=$th4;}
elseif ($table === "ts") {$stds=$td5;$ths=$th5;}
elseif ($table === "status_maintenance") {$stds=$td6;$ths=$th6;}
elseif ($table === "responsible_for_commissioning") {$stds=$td7;$ths=$th7;}
elseif ($table === "location") {$stds=$td8;$ths=$th8;}

?>
<!DOCTYPE html>
<html lang="ru">
<head>
<meta charset="UTF-8">
```

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

ДП-02069964-090203-07-18

Лист

72

Продолжение ПРИЛОЖЕНИЯ В

```

        <input type="text" class="form-control pull-left" id="search"
placeholder="Поиск по таблице">
        <div id="printTable">
        <style type="text/css">
            th,
            td
            {
            padding: 4px 4px 4px 4px ;
            }
            th
            {
            border-top: 2px solid #333333 ;
            border-bottom: 2px solid #333333 ;
            }
            td
            {
            border-bottom: 1px dotted #999999 ;
            }
        </style>
        <table id="mytable">
        <tr>
            <? foreach ($ths as $th) {
            echo "<th>".$th."</th>";
            }
            if ($_SESSION['flag'] <> 'b') {
            echo "<th>Изменить</th>";
            echo "<th>Удалить</th>";
            }
        ?>
        </tr>
        <?foreach ($getRows as $Row) {
            echo "<tr>";
            foreach ($tds as $td) {
            echo "<td>".$Row[$td]."</td>";
            }
            if ($_SESSION['flag'] <> 'b') {
            echo '<td><a href="edit.php?id='.$Row['id'].'">Изменить</a></td>';
            echo '<td><a href="delete.php?id='.$Row['id'].'">Удалить</a></td>';
            }
        }
        echo "</tr>";
        } ?>
        </table>

        </div>

        </div>
        </div>
        </section>
        <script>
$(document).ready(function(){
    $("#search").keyup(function(){
        _this = this;
        $.each($("#mytable tbody tr"), function() {
            if($(this).text().toLowerCase().indexOf($_this.val().toLowerCase()) === -1)
                $(this).hide();
            else

```

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

ДП-02069964-090203-07-18

Лист

74

Продолжение ПРИЛОЖЕНИЯ В

```
        $(this).show();
    });
});
</script>
<script type="text/javascript">
    function printData()
    {
        var divToPrint=document.getElementById("printTable");
        newWin= window.open("");
        newWin.document.write(divToPrint.outerHTML);
        newWin.print();
        newWin.close();
    }

$('button').on('click',function(){
printData();
})
</script>
</body>
</html>
```

10 Файл «add.php» – реализация ввода информации в базу данных

```
<?
session_start();
if (!isset($_SESSION['table'])) {
header("Location: /service.php");
}
unset($table);
require_once 'config.php';
$table = $_SESSION['table'];
$db = new Database();
if ($table === "users") {
    $ths = [
        "Логин" => "Логин", "Пароль" => "Пароль", "Роль" => "Роль"
    ];
if (isset($_POST['frizzes'])) {
    $text1 = $_POST['text1'];
    $text2 = $_POST['text2'];
    $text3 = $_POST['text3'];
    if ($text3 === "внутренние службы"){ $flag = "a"; }
    elseif($text3 === "сторонние службы"){ $flag = "b"; }
    elseif($text3 === "администратор системы"){ $flag = "c"; }
    elseif($text3 === "администратор предприятия"){ $flag = "d"; }
    else{ $flag = "a"; }
    $insertRow = $db->insertRow("INSERT INTO users(login, password, position, flag) VALUE(?, ?, ?, ?)",
[ $text1, $text2, $text3, $flag]);
        unset($_SESSION['table']);
        unset($_POST['frizzes']);
        header("Location: /table.php?table=$table");
    }
}
elseif ($table === "request") {
    $ths = [
        "Имя" => "Имя",
```

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

ДП-02069964-090203-07-18

Лист

75

Продолжение ПРИЛОЖЕНИЯ В

```
"Email" => "Email",
"Текст" => "Текст"
];
if (isset($_POST['frizzes'])) {
    $text1 = $_POST['text1'];
    $text2 = $_POST['text2'];
    $text3 = $_POST['text3'];
    $insertRow = $db->insertRow("INSERT INTO request(name, email, txt) VALUE(?, ?, ?)", [$text1, $text2,
$text3]);
                                unset($_SESSION['table']);
                                unset($_POST['frizzes']);
    header("Location: /table.php?table=$table");
}
}
elseif ($table === "technical_mean") {
    $ths = [
        "Наименование" => "Наименование", "Модель" => "Модель", "Серийный номер" => "Серийный
номер", "Инвентарный номер" => "Инвентарный номер", "Год выпуска" => "Год выпуска", "Дата ввода в
эксплуатацию" => "Дата ввода в эксплуатацию", "Параметр" => "Параметр", "Параметр" =>
"Параметр", "Параметр" => "Параметр", "Параметр" => "Параметр", "Параметр" => "Параметр", "Параметр" =>
"Параметр", "Параметр" => "Параметр", "Параметр" => "Параметр", "Параметр" => "Параметр", "Статус" =>
"Статус",
    ];
    if (isset($_POST['frizzes'])) {
        $text1 = $_POST['text1'];
        $text2 = $_POST['text2'];
        $text3 = $_POST['text3'];
        $text4 = $_POST['text4'];
        $text5 = $_POST['text5'];
        $text6 = $_POST['text6'];
        $text7 = $_POST['text7'];
        $text8 = $_POST['text8'];
        $text9 = $_POST['text9'];
        $text10 = $_POST['text10'];
        $text11 = $_POST['text11'];
        $text12 = $_POST['text12'];
        $text13 = $_POST['text13'];
        $text14 = $_POST['text14'];
        $text15 = $_POST['text15'];
        $text16 = $_POST['text16'];
        $insertRow = $db->insertRow("INSERT INTO technical_mean(name, model, serial_number,
inventory_number, release_year, start_exploitation, option1, option2, option3, option4, option5, option6, option7,
option8, option9, status) VALUE(?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)", [$text1, $text2, $text3, $text4, $text5, $text6,
$text7, $text8, $text9, $text10, $text11, $text12, $text13, $text14, $text15, $text16]);
                                unset($_SESSION['table']);
                                unset($_POST['frizzes']);
        header("Location: /table.php?table=$table");
    }
}
elseif ($table === "plan_maintenance") {
    $ths = [
        "Технические средства" => "Технические средства",
        "Дата" => "Дата",
        "Статус" => "Статус",
    ];
    if (isset($_POST['frizzes'])) {
        $text1 = $_POST['text1'];
```

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

ДП-02069964-090203-07-18

Лист

76

Продолжение ПРИЛОЖЕНИЯ В

```
$text2 = $_POST['text2'];
$text3 = $_POST['text3'];
$insertRow = $db->insertRow("INSERT INTO plan_maintenance(technical_means, date_plan, status)
VALUE(?, ?, ?)", [$text1, $text2, $text3]);
unset($_SESSION['table']);
unset($_POST['frizzes']);
header("Location: /table.php?table=$table");
}
}
elseif ($table === "location") {
    $ths = [
        "Филиал" => "Филиал",
        "Адрес" => "Адрес",
    ];
    if (isset($_POST['frizzes'])) {
        $text1 = $_POST['text1'];
        $text2 = $_POST['text2'];
        $insertRow = $db->insertRow("INSERT INTO location(branch, address) VALUE(?, ?)", [$text1, $text2]);
        unset($_SESSION['table']);
        unset($_POST['frizzes']);
        header("Location: /table.php?table=$table");
    }
}
elseif ($table === "responsible_for_commissioning") {
    $ths = [
        "ФИО" => "ФИО",
        "Профессия" => "Профессия",
        "Место работы" => "Место работы",
    ];
    if (isset($_POST['frizzes'])) {
        $text1 = $_POST['text1'];
        $text2 = $_POST['text2'];
        $text3 = $_POST['text3'];
        $insertRow = $db->insertRow("INSERT INTO responsible_for_commissioning(full_name, profession, job)
VALUE(?, ?, ?)", [$text1, $text2, $text3]);
        unset($_SESSION['table']);
        unset($_POST['frizzes']);
        header("Location: /table.php?table=$table");
    }
}
elseif ($table === "status_maintenance") {
    $ths = [
        "Статус" => "Статус",
    ];
    if (isset($_POST['frizzes'])) {
        $text1 = $_POST['text1'];
        $insertRow = $db->insertRow("INSERT INTO status_maintenance(status) VALUE(?)", [$text1]);
        unset($_SESSION['table']);
        unset($_POST['frizzes']);
        header("Location: /table.php?table=$table");
    }
}
elseif ($table === "ts") {
    $ths = [
        "Наименование" => "Наименование", "Параметр" => "Параметр",
    ];
    if (isset($_POST['frizzes'])) {
```

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

ДП-02069964-090203-07-18

Лист

77

Продолжение ПРИЛОЖЕНИЯ В

```
$text1 = $_POST['text1'];
$text2 = $_POST['text2'];
$insertRow = $db->insertRow("INSERT INTO ts(name, option1) VALUE(?, ?)", [$text1, $text2]);
        unset($_SESSION['table']);
        unset($_POST['frizzes']);
    header("Location: /table.php?table=$table");
}
}
elseif ($table === "workstation") {
    $ths = [
        "Рабочая станция" => "Рабочая станция",
    ];
    if (isset($_POST['frizzes'])) {
        $text1 = $_POST['text1'];
        $insertRow = $db->insertRow("INSERT INTO workstation(workstation) VALUE(?)", [$text1]);
        unset($_SESSION['table']);
        unset($_POST['frizzes']);
        header("Location: /table.php?table=$table");
    }
}
?>
<!DOCTYPE html>
<html lang="ru">
    <head>
        <meta charset="UTF-8">
        <title>Редактор</title>
        <meta http-equiv="content-type" content="text/html; charset=utf-8" />
        <!--[if lte IE 8]><script src="js/html5shiv.js"></script><![endif]-->
        <script src="js/jquery.min.js"></script>
        <script src="js/skel.min.js"></script>
        <script src="js/skel-layers.min.js"></script>
        <script src="js/init.js"></script>
        <noscript>
            <link rel="stylesheet" href="css/skel.css" />
            <link rel="stylesheet" href="css/style.css" />
            <link rel="stylesheet" href="css/style-xlarge.css" />
        </noscript>
    </head>
    <body>
        <!-- Header -->
        <header id="header">
            <nav id="nav">
                <ul>
                    <li><a href="index.php">Главная</a></li>
                    <li><a href="request.php">Подать заявку</a></li>
                    <li><a href="service.php" class="button special">Сервис
служб</a></li>
                </ul>
            </nav>
        </header>
        <!-- One -->
        <center>
            <section id="main" class="wrapper">
                <div class="container">
                    <form method="post" action='add.php'>
                        <div>
                            <div class="6u 12u$(4)">
```

Продолжение ПРИЛОЖЕНИЯ В

```
id='login' value=" ".$th.">";

        <h3>Добавить</h3>
        <? $i = "1";
        foreach ($ths as $th){
            echo "<input type='text' name='text' ".$i."

        $i++;
        }

        ?>
        </div>
        <div class="12u$">
            <ul class="actions">
                <li><input type="submit" name="frizzes"
value="Добавить" class="special" /></li>

                <li><input type="reset" value="Отмена" /></li>
            </ul>
        </div>
    </div>
</form>
</div>
</section></center><br />
</body>
</html>
```

11 Файл «update.php» – изменение учётных данных

```
<?
session_start();
if (!isset($_SESSION['login'])) {
header("Location: /auth.php");
}
if (!isset($_SESSION['true'])) {
header("Location: /index.php");
}
unset($_SESSION['true']);
require_once 'config.php';
$db = new Database();
if (isset($_POST['submit'])) {
    if ( $_POST['password'] <> null && $_POST['login'] <> null ) {
        $password = $_POST['password'];
        $login = $_POST['login'];
        $id = $_SESSION['id'];
        $updateRow = $db->updateRow("UPDATE users SET password = ?, login = ? WHERE id = ?", ["$password",
"$login", "$id"]);
        $db->Disconnect();
        $_SESSION['login'] = $login;
        header("Location: /service.php");
    }
    else{
        header("Location: /service.php");
    }
}
?>
```

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

ДП-02069964-090203-07-18

Лист

79

Продолжение ПРИЛОЖЕНИЯ В

```
if (isset($_POST['submit'])) {
    $post = $_POST['submit'];
    $text1 = $_POST['text1'];
    $text2 = $_POST['text2'];
    $text3 = $_POST['text3'];
    $update = $edit->$table($db,$id,$post,$text1,$text2,$text3);
        unset($_SESSION['table']);
    header("Location: /table.php?table=$table");
}
}
elseif ($table === "workstation") {
    $getRow = $db->getRow("SELECT workstation, component1, component2 FROM $table WHERE id
= ?", ["$id"]);
if (isset($_POST['submit'])) {
    $post = $_POST['submit'];
    $text1 = $_POST['text1'];
    $text2 = $_POST['text2'];
    $text3 = $_POST['text3'];
    $update = $edit->$table($db,$id,$post,$text1,$text2,$text3);
        unset($_SESSION['table']);
    header("Location: /table.php?table=$table");
}
}
elseif ($table === "ts") {
    $getRow = $db->getRow("SELECT name FROM $table WHERE id = ?", ["$id"]);
if (isset($_POST['submit'])) {
    $post = $_POST['submit'];
    $text1 = $_POST['text1'];
    $update = $edit->$table($db,$id,$post,$text1);
        unset($_SESSION['table']);
    header("Location: /table.php?table=$table");
}
}
elseif ($table === "status_maintenance") {
    $getRow = $db->getRow("SELECT status FROM $table WHERE id = ?", ["$id"]);
if (isset($_POST['submit'])) {
    $post = $_POST['submit'];
    $text1 = $_POST['text1'];
    $update = $edit->$table($db,$id,$post,$text1);
        unset($_SESSION['table']);
    header("Location: /table.php?table=$table");
}
}
elseif ($table === "responsible_for_commissioning") {
    $getRow = $db->getRow("SELECT full_name, profession, job FROM $table WHERE id = ?",
["$id"]);
if (isset($_POST['submit'])) {
    $post = $_POST['submit'];
    $text1 = $_POST['text1'];
    $text2 = $_POST['text2'];
    $text3 = $_POST['text3'];
    $update = $edit->$table($db,$id,$post,$text1,$text2,$text3);
        unset($_SESSION['table']);
    header("Location: /table.php?table=$table");
}
}
}
```

Изм.	Лист	№ докум.	Подпись	Дата

ДП-02069964-090203-07-18

Лист

81

Продолжение ПРИЛОЖЕНИЯ В

```
elseif ($table === "location") {
    $getRow = $db->getRow("SELECT branch, address FROM $table WHERE id = ?", ["$id"]);
if (isset($_POST['submit'])) {
    $post = $_POST['submit'];
    $text1 = $_POST['text1'];
    $text2 = $_POST['text2'];
    $update = $edit->$table($db,$id,$post,$text1,$text2);
        unset($_SESSION['table']);
    header("Location: /table.php?table=$table");
}
}

?>
<!DOCTYPE html>
<html lang="ru">
    <head>
        <meta charset="UTF-8">
        <title>Редактор</title>
        <meta http-equiv="content-type" content="text/html; charset=utf-8" />
        <!--[if lte IE 8]><script src="js/html5shiv.js"></script><![endif]-->
        <script src="js/jquery.min.js"></script>
        <script src="js/skel.min.js"></script>
        <script src="js/skel-layers.min.js"></script>
        <script src="js/init.js"></script>
        <noscript>
            <link rel="stylesheet" href="css/skel.css" />
            <link rel="stylesheet" href="css/style.css" />
            <link rel="stylesheet" href="css/style-xlarge.css" />
        </noscript>
    </head>
    <body>

    <!-- Header -->
        <header id="header">
            <nav id="nav">
                <ul>
                    <li><a href="index.php">Главная</a></li>
                    <li><a href="request.php">Подать заявку</a></li>
                    <li><a href="service.php" class="button special">Сервис служб</a></li>
                </ul>
            </nav>
        </header>

    <!-- One -->
    <center>
        <section id="main" class="wrapper">
            <div class="container">
                <form method="post" action='edit.php?id=<? echo $id?>'>
                    <div>
                        <div class="6u 12u$(4)">
                            <h3>Редактировать</h3>
                            <? $i = "1"; foreach ($getRow as $row) {
                                echo "<input type='text' name='text' .$.i.'" id='login'
value=" . $row.">";
                                    $i++;
                                }
                            ?>
                        <br />
                    </div>
                </form>
            </div>
        </section>
    </center>

```

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

ДП-02069964-090203-07-18

Лист

82

Продолжение ПРИЛОЖЕНИЯ В

```
        </div>
        <div class="12u$">
            <ul class="actions">
                <li><input type="submit" name="submit" value="Изменить"
class="special" /></li>
                <li><input type="reset" value="Отмена" /></li>
            </ul>
        </div>
    </div>
</form>
</div>
</section></center><br />
</body>
</html>
```

13 Файл «function.php» – реализация функции обновления таблиц базы данных

```
<?php
class Edit
{
    public function users($db,$id,$post,$text1,$text2,$text3){
        if ( $text1 <> null && $text1 <> null && $text3 <> null) {
            $updateRow = $db->updateRow("UPDATE users SET login = ?, password = ?, position = ? WHERE id = ?",
[ $text1, $text2, $text3, "$id"]);
            return;
        }
    }
    public function workstation($db,$id,$post,$text1,$text2,$text3){
        if ( $text1 <> null && $text1 <> null && $text3 <> null) {
            $updateRow = $db->updateRow("UPDATE workstation SET workstation = ?, component1 = ?, component2 = ?
WHERE id = ?", [ $text1, $text2, $text3, "$id"]);
            return;
        }
    }
    public function ts($db,$id,$post,$text1){
        if ( $text1 <> null) {
            $updateRow = $db->updateRow("UPDATE ts SET name = ? WHERE id = ?", [ $text1, "$id"]);
            return;
        }
    }
    public function status_maintenance($db,$id,$post,$text1){
        if ( $text1 <> null) {
            $updateRow = $db->updateRow("UPDATE status_maintenance SET status = ? WHERE id = ?", [ $text1, "$id"]);
            return;
        }
    }

    public function responsible_for_commissioning($db,$id,$post,$text1,$text2,$text3){
        if ( $text1 <> null && $text1 <> null && $text3 <> null) {
            $updateRow = $db->updateRow("UPDATE responsible_for_commissioning SET full_name = ?, profession = ?,
job = ? WHERE id = ?", [ $text1, $text2, $text3, "$id"]);
            return;
        }
    }
}
```

										Лист
										83
Изм.	Лист	№ докум.	Подпись	Дата	ДП-02069964-090203-07-18					

Окончание ПРИЛОЖЕНИЯ В

```
    }
}
public function location($db,$id,$post,$text1,$text2){
    if ( $text1 <> null && $text1 <> null) {
        $updateRow = $db->updateRow("UPDATE location SET branch = ?, address = ? WHERE id = ?", [$text1, $text2,
"$id"]);
    }
    return;
}

    public function technical_mean($db,$id,$post,$text1,$text2,$text3,$text4,$text5){
    if ( $text1 <> null && $text1 <> null && $text3 <> null) {
        $updateRow = $db->updateRow("UPDATE technical_mean SET name = ?, model = ?, serial_number = ?,
inventory_number = ?, release_year = ?, start_exploitation = ?, option1 = ?, option2 = ?, option3 = ?, option4 = ?,
option5 = ?, option6 = ?, option7 = ?, option8 = ?, option9 = ?, status = ? WHERE id = ?", [$text1, $text2, $text3,
$text4, $text5, $text6, $text7, $text8, $text9, $text10, $text11, $text12, $text13, $text14, $text15, $text16, "$id"]);
    }
    return;
}

    public function plan_maintenance($db,$id,$post,$text1,$text2,$text3){
    if ( $text1 <> null && $text1 <> null && $text3 <> null) {
        $updateRow = $db->updateRow("UPDATE plan_maintenance SET technical_means = ?, date_plan = ?, status = ?
WHERE id = ?", [$text1, $text2, $text3, "$id"]);
    }
    return;
}

}
?>
```

14 Файл «delete.php» – удаление записи таблицы

```
<?
session_start();
if (!isset($_SESSION['table'])) {
    header("Location: /service.php");
}
require_once 'config.php';
$db = new Database();
if (isset($_GET['id'])) {
    $table = $_SESSION['table'];
    $id = $_GET['id'];
    $deleteRow = $db->deleteRow("DELETE FROM $table WHERE id = ?", ["$id"]);
    $db->Disconnect();
    unset($_SESSION['table']);
    header("Location: /table.php?table=$table");
}
?>
```

										Лист
										84
Изм.	Лист	№ докум.	Подпись	Дата	ДП-02069964-090203-07-18					

ПРИЛОЖЕНИЕ Г

(справочное)

Внешний вид интерфейса

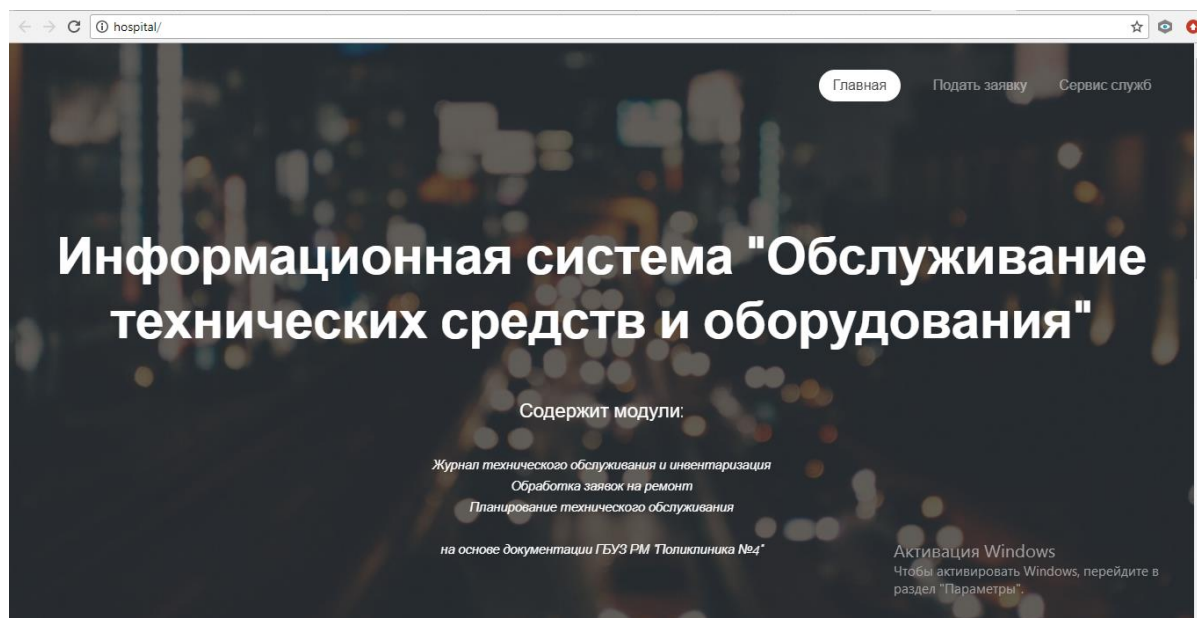


Рисунок Г.1 – Интерфейс главной страницы

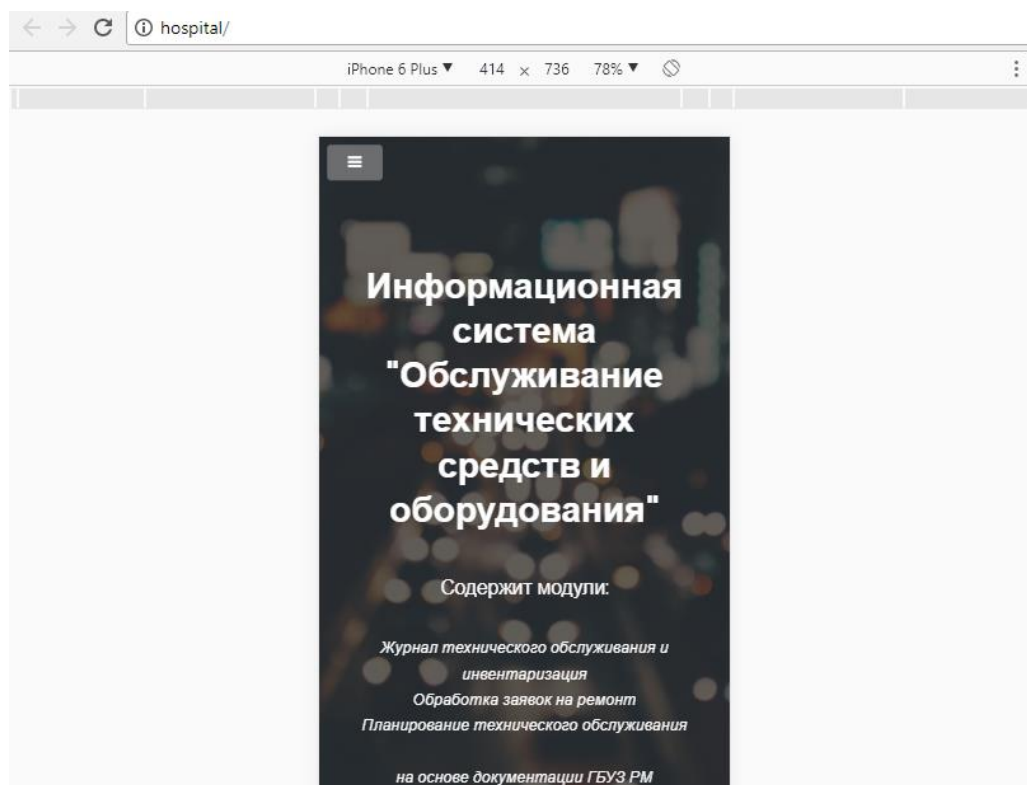


Рисунок Г.2 – Оптимизация под мобильные устройства

Изм.	Лист	№ докум.	Подпись	Дата

ДП-02069964-090203-07-18

Лист

85

Продолжение ПРИЛОЖЕНИЯ Г

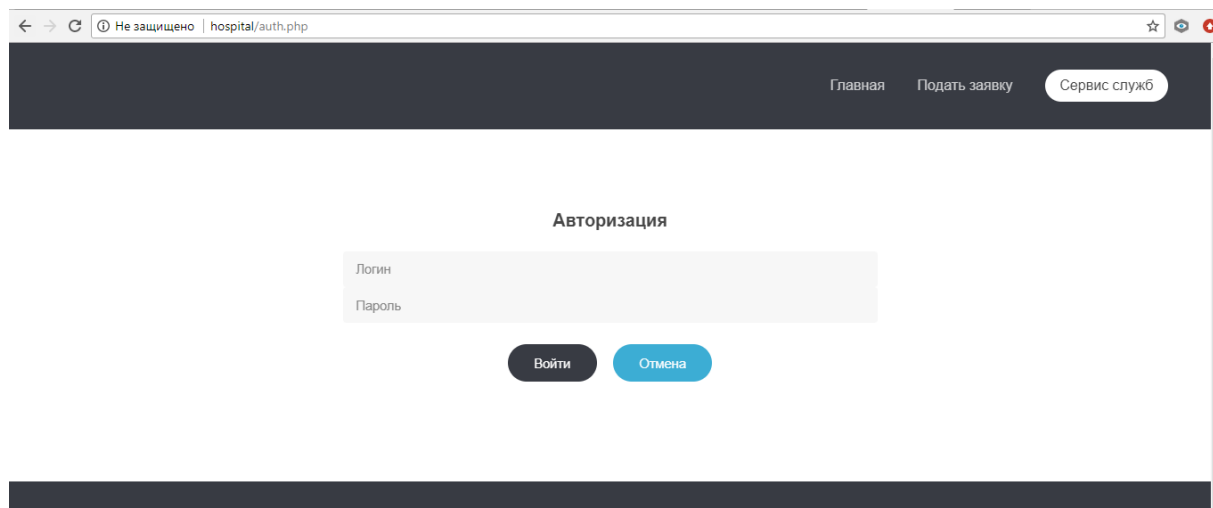


Рисунок Г.3 – Интерфейс авторизации пользователя

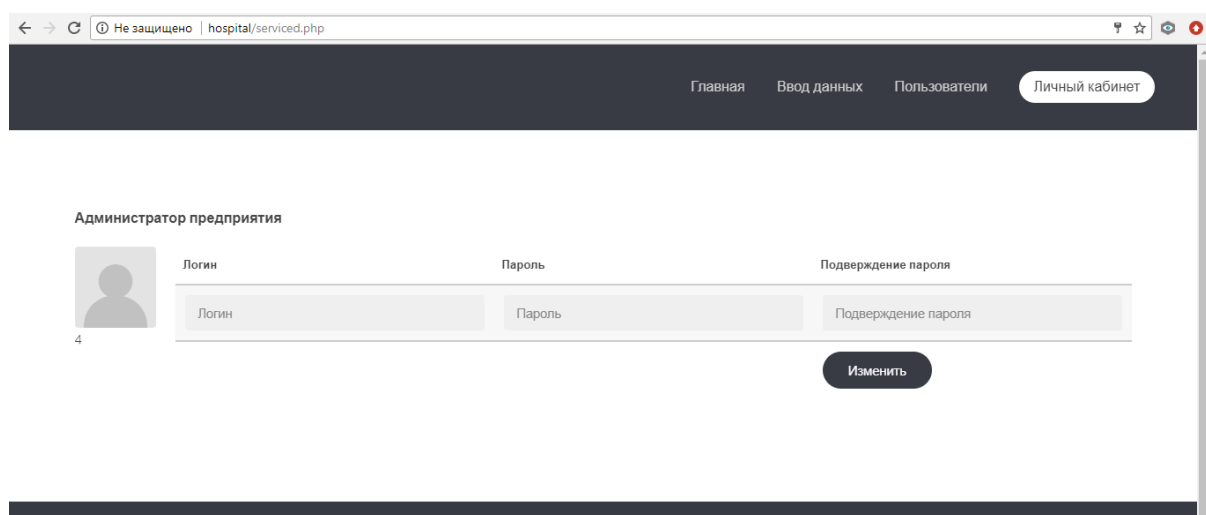


Рисунок Г.4 – Интерфейс личного кабинета пользователя

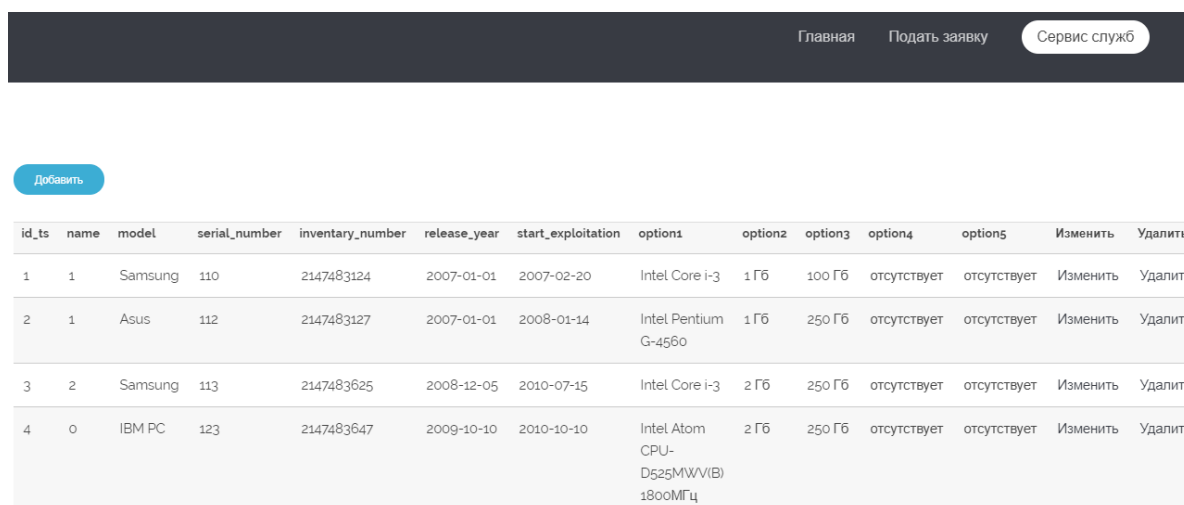


Рисунок Г.5 – Интерфейс формы вывода

Окончание ПРИЛОЖЕНИЯ Г

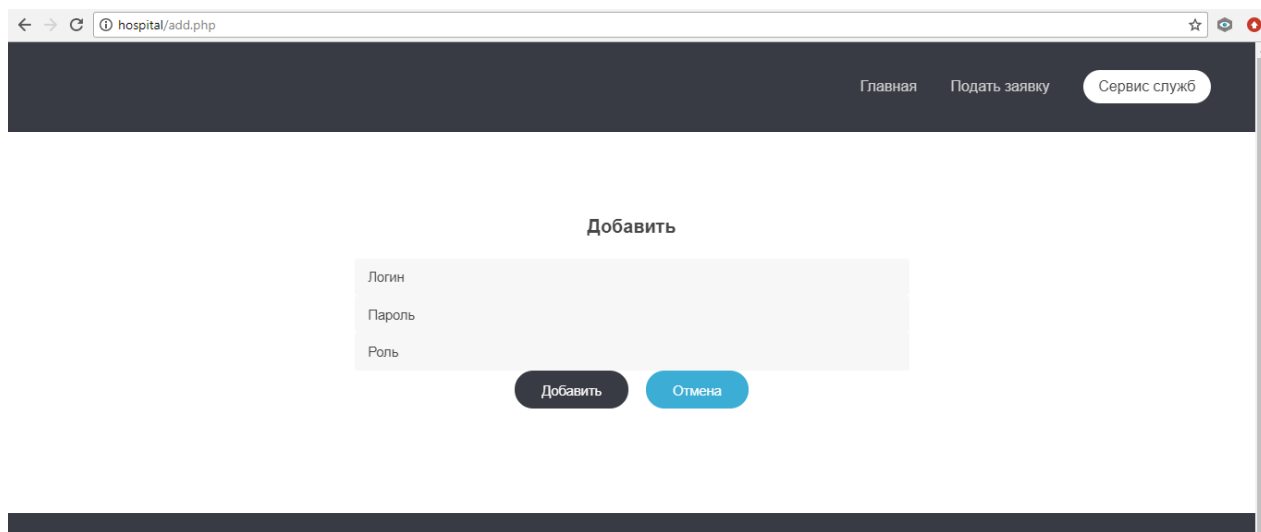


Рисунок Г.6 – Интерфейс вводной компоненты пользователей

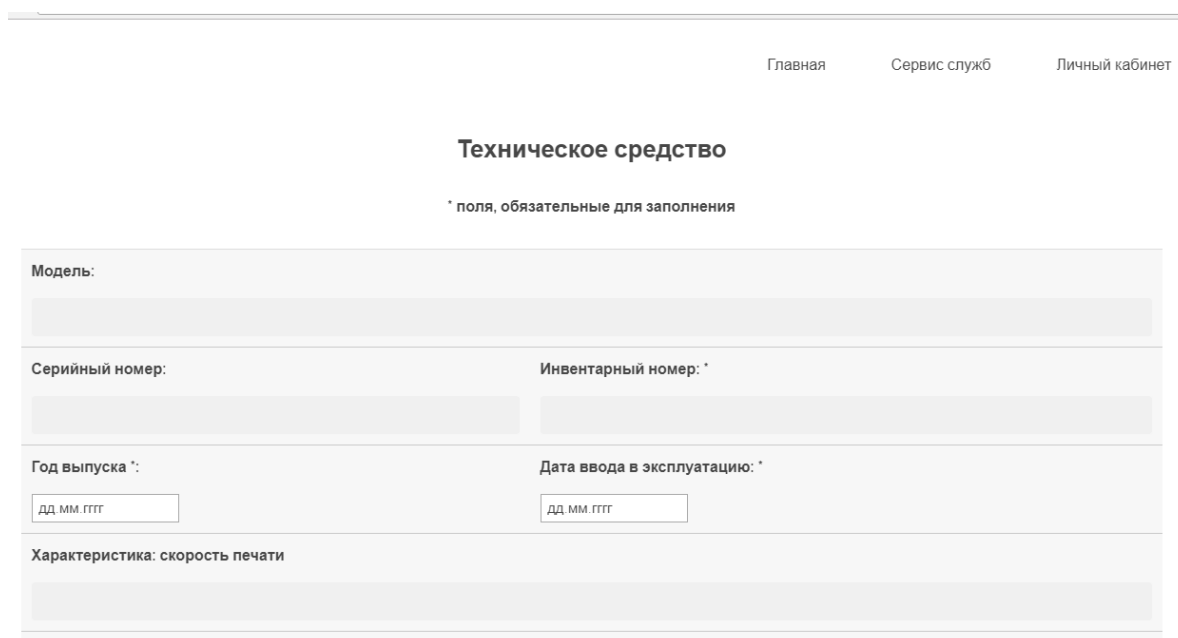


Рисунок Г.7 – Интерфейс вводной компоненты технических средств

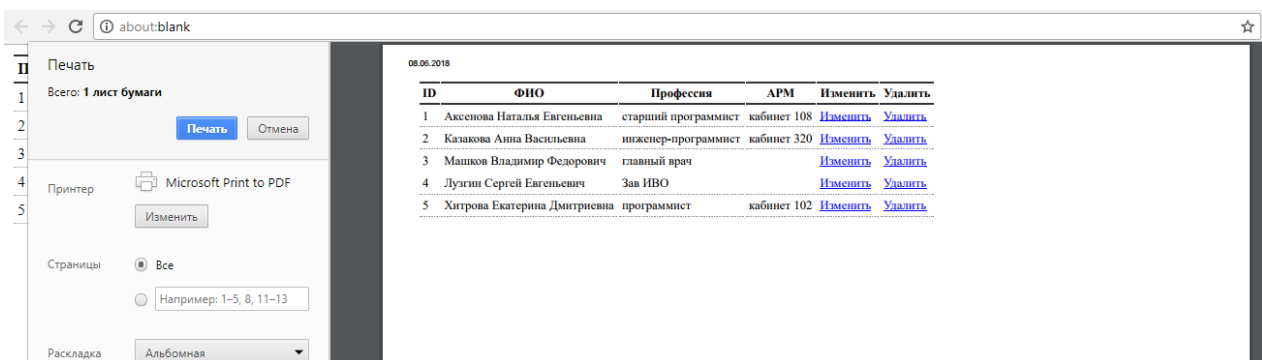


Рисунок Г.8 – Вывод на печать

ОТЧЕТ

о результатах проверки работы обучающегося
на наличие заимствований

Ф.И.О. автора работы Камаева Анастасия Андреевна

Тема работы Разработка программного модуля «Журнал технического обслуживания и инвентаризация» информационной системы «Обслуживание технических средств и оборудования» (на примере ГБУЗ РМ «Поликлиника №4»)

Руководитель Лузгин Сергей Евгеньевич

Представленная работа прошла проверку на наличие заимствований в системе «Антиплагиат.ВУЗ»

Результат автоматической проверки: оригинальность 85,29%
цитирования 5,80%
заимствования 8,91%

Результат анализа полного отчета на наличие заимствований:

правоверные заимствования:	8,91%	да/нет, количество (%), обоснованность
корректные цитирования:	5,80%	да/нет, количество (%), обоснованность
неправомерные заимствования:	нет	да/нет, количество (%), обоснованность
признаки обхода системы:	нет	да/нет, описание

Общее заключение об итоговой оригинальности работы и возможности ее допуска к защите: допустить к защите.

Руководитель
преподаватель ВКК



С.Е. Лузгин



ВСЕРОССИЙСКИЙ ОБРАЗОВАТЕЛЬНЫЙ

ПОРТАЛ ПЕДАГОГА

СВИДЕТЕЛЬСТВО О ПУБЛИКАЦИИ

Настоящим свидетельством подтверждается, что

**студент(ка)
ФГБОУ ВО «НИ МГУ им. Н.П. Огарёва»
г. Саранска**

Камаева Анастасия Андреевна

***опубликовал(а) на официальном сайте
Всероссийского издания "Портал педагога"
учебно-методический материал***

*Наименование материала: общеисследовательская статья
Тема: Роль самостоятельной работы студентов среднего
профессионального образования в образовательном процессе*

*Веб-адрес размещения публикации
<https://portalpedagoga.ru/servisy/publik/publ?id=31783>*

Главный редактор



В.В.Богданов

Всероссийский образовательный «Портал Педагога». Доменное имя в сети интернет portalpedagoga.ru. Свидетельство о регистрации СМИ Эл № ФС 77 - 65786. Выдано 20.05.2016г. Федеральной службой России по надзору в сфере связи, информационных технологий и массовых коммуникаций. Свидетельство о публикации серия АА №31783 от 11.06.2018 г.



СПРАВКА

о результатах проверки текстового документа на наличие заимствований

Проверка выполнена в системе Антиплагиат.ВУЗ

Автор работы	Камаева Анастасия Андреевна
Факультет, кафедра, номер группы	ФДП и СПО
Тип работы	Выпускная квалификационная работа
Название работы	Разработка программного модуля _Журнал технического обслуживания и инвентаризация_ информационной системы _Обслуживание технических средств и оборудования_ (на примере ГБУЗ РМ _Поликлиника _4_)
Название файла	Камаева_диплом.docx
Процент заимствования	8,91%
Процент цитирования	5,80%
Процент оригинальности	85,29%
Дата проверки	13:50:01 09 июня 2018г.
Модули поиска	Модуль выделения библиографических записей; Сводная коллекция ЭБС; Коллекция РГБ; Цитирования; Переводные заимствования; Коллекция Гарант; Модуль поиска Интернет; Модуль поиска "МГУ им. Н. П. Огарева"; Модуль поиска общеупотребительных выражений; Кольцо Вузов
Работу проверил	ПЕТЯНКИН МАКСИМ ФЕДОРОВИЧ ФИО проверяющего
Дата подписи	09.06.18  Подпись проверяющего

Чтобы убедиться
в подлинности справки,
используйте QR-код, который
содержит ссылку на отчет.



Ответ на вопрос, является ли обнаруженное заимствование
корректным, система оставляет на усмотрение проверяющего.
Предоставленная информация не подлежит использованию
в коммерческих целях.

ОТЗЫВ РУКОВОДИТЕЛЯ ДИПЛОМНОГО ПРОЕКТА

Тема работы Разработка программного модуля «Журнал технического обслуживания и инвентаризация» информационной системы «Обслуживание технических средств и оборудования» (на примере ГБУЗ РМ «Поликлиника №4»).

Автор (студент) Камаева Анастасия Андреевна.

Факультет довузовской подготовки и среднего профессионального образования.

Выпускающая предметная цикловая комиссия (кафедра) общепрофессиональных специальных (информационно-коммуникационных) дисциплин.

Специальность 09.02.03 Программирование в компьютерных системах.

Руководитель С.Е. Лузгин, преподаватель высшей квалификационной категории.

Оценка соответствия требованиям ФГОС СПО подготовленности автора выпускной работы

Требования к профессиональной подготовке	Соответствует	В основном соответствует	Не соответствует
Уметь корректно формулировать и ставить задачи (проблемы) своей деятельности при выполнении дипломной работы, анализировать, диагностировать причины появления проблем, их актуальность	*		
Устанавливать приоритеты и методы решения поставленных задач (проблем)	*		
Уметь использовать информацию – правильно оценить и обобщить степень изученности объекта исследования (в соответствии со специализацией)	*		
Владеть методами сбора, хранения и обработки (редактирования) информации, применяемой в сфере профессиональной деятельности (по специализации)	*		
Владеть современными методами анализа и интерпретации полученной информации, оценивать их возможности при решении поставленных задач (по специализации)	*		
Уметь рационально планировать время выполнения работы, определять грамотную последовательность и объем операций и решений при выполнении поставленной задачи	*		
Уметь объективно оценивать полученные результаты расчетов, вычислений, используя для сравнения данные других направлений	*		
Уметь анализировать полученные результаты интерпретации данных (в рамках специализации)	*		
Знать методы системного анализа	*		
Уметь осуществлять деятельность в кооперации с коллегами, находить компромиссы при совместной деятельности	*		
Уметь делать самостоятельные обоснованные и достоверные выводы из проделанной работы	*		
Уметь пользоваться научной литературой профессиональной направленности	*		

Отмеченные достоинства: Актуальность проекта обусловлена необходимостью создания информационной системы предназначенной для автоматизации работы техников и инженеров, занимающихся обслуживанием технических средств, оборудования и программного обеспечения, поскольку это позволяет планировать и систематизировать работу соответствующих специалистов, упорядочить информацию об эксплуатации имеющихся технических средств. В первой главе выполнен комплекс работ, направленных на анализ предметной области: определена сущность задачи, произведено планирование разработки. Во второй главе ведется подробное описание процесса разработки программного модуля «Журнал технического обслуживания и инвентаризация». В третьей главе дипломного проекта обозначены реализация и последующее тестирование разработанного продукта.

Выполненный дипломный проект представляет собой программный модуль, реализованный при помощи веб-технологий с удобным и интуитивно понятным графическим интерфейсом пользователя, полностью готовый к применению. В процессе реализации данного проекта проведен анализ предметной области, осуществлен сбор необходимой информации; обеспечено корректное отображение сайта в популярных браузерах; произведено тестирование полученной системы.

Отмеченные недостатки:

Ввиду отсутствия ряда модулей информационной системы «Обслуживание технических средств и оборудования» проект находится на стадии отладки взаимодействия с другими модулями информационной системы.

Заключение: оценивая в целом дипломный проект Камаевой Анастасии Андреевны, следует отметить, что она отвечает основным требованиям и может быть допущена к защите. Оригинальность текста составляет 85,29%. Автор заслуживает присвоения квалификации техник-программист.

Руководитель _____


(подпись)

« 25 »

мая

2018 г.

Рецензия

на выпускную квалификационную работу (проект),
выполненную студенткой факультета довузовской подготовки и среднего
профессионального образования, 4 курса, специальности «Программирование в
компьютерных системах»

Камаевой Анастасией Андреевной,
на тему: «Разработка программного модуля «Журнал технического обслуживания и
инвентаризации» информационной системы «Обслуживание технических средств и
оборудования» (на примере ГБУЗ РМ «Поликлиника №4»)»

На рецензию представлен дипломный проект, в составе программного модуля
«Журнал технического обслуживания и инвентаризации» информационной системы
«Обслуживание технических средств и оборудования» и пояснительной записки,
выполненной на 55 листах.

Актуальность проекта обусловлена необходимостью создания
информационной системы предназначенной для автоматизации работы техников и
инженеров, занимающихся обслуживанием технических средств, оборудования и
программного обеспечения, поскольку это позволяет планировать и
систематизировать работу соответствующих специалистов, упорядочить
информацию об эксплуатации имеющихся технических средств.

Качество дипломного проекта соответствует всем техническим требованиям,
оформление интерфейса программного продукта интуитивно понятно и удобно в
использовании.

В соответствии с заданием на дипломный проект студенткой Камаевой А. А. в
первой главе пояснительной записки был проведен анализ предметной области,
методов и средств создания программного модуля. Во второй главе описан процесс
разработки программного модуля. В третьей главе описаны реализованные
компоненты модуля и процесс тестирования на наличие ошибок.

Представленный дипломный проект выполнен технически грамотно в
соответствии с требованиями ГОСТа.

Достоинствами проекта являются доступный для восприятия интерфейс,
корректная работа функций инвентаризации технических средств и назначения
ответственных лиц, быстрая загрузка форм интерактивного интерфейса.

Существенных недостатков в дипломном проекте выявлено не было. Несущественным недостатком является лишь то, что сайт не адаптирован под браузер Internet Explorer ниже версии Internet Explorer 8 по причине отсутствия настроек CSS для адаптивности. Однако отмеченный недостаток не снижает общего уровня проделанной работы над проектом в целом.

Оценивая в целом дипломный проект Камаевой Анастасии Андреевны, следует отметить, что она соответствует требованиям, предъявляемым к дипломным работам, может быть допущена к защите и заслуживает оценки отлично.

Рецензент:

Инженер-программист
информационно-аналитического отдела
ГБУЗ РМ «Поликлиника №4»

А. В. Казакова 04.06.18
(дата) (подпись)



А. В. Казакова