

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ КАЗАХСТАН

Satbayev University

Институт кибернетики и информационных технологий

Кафедра кибербезопасность, обработка и хранение информации

Ахмет Еркебұлан Әділбекұлы

Разработка онлайн гардероба

ДИПЛОМНАЯ РАБОТА

Специальность 5В070300 – Информационные системы

Алматы, 2020

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ КАЗАХСТАН

Satbayev University

Институт кибернетики и информационных технологий

Кафедра кибербезопасность, обработка и хранение информации

ДОПУЩЕН К ЗАЩИТЕ

Заведующий кафедрой _____

(наименование кафедры)

(ученая степень, звание)

подпись _____ Ф.И.О.

« _____ » _____ 20__ г.

ДИПЛОМНАЯ РАБОТА

На тему: “Разработка онлайн гардероба”

Специальность 5В070300 – Информационные системы

Выполнил:

Ахмет Е. Ә.

Рецензент

Научный руководитель

« _____ » _____ 20__ г.

_____ Иманбаев А. Ж.

« _____ » _____ 20__ г.

Алматы, 2020

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Satbayev University

Институт кибернетики и информационных технологий

Кафедра кибербезопасность, обработка и хранение информации

5B070300 – Информационные системы

УТВЕРЖДАЮ

Заведующий кафедрой _____

(наименование кафедры)

(ученая степень, звание)

подпись

Ф.И.О.

« _____ » _____ 20__ г.

ЗАДАНИЕ

на выполнение дипломной работы

Обучающемуся Ахмет Еркебулан Әділбекұлы

Тема: Разработка онлайн гардероба

Утверждена приказом Ректора Университета № 762–б от "27" января 2020 г.

Срок сдачи законченной работы "27" мая 2020 г.

Исходные данные к дипломной работе: результат обзора современного состояния по данной теме; сбор теоретического материала.

Краткое содержание дипломной работы:

- а) изучение современного состояния поставленной задачи;
- б) разработка веб-приложения;

Рекомендуемая основная литература: *из 17 наименований*

ГРАФИК

подготовки дипломной работы (проекта)

Наименование разделов, перечень разрабатываемых вопросов	Сроки представления научному руководителю	Примечание
Обзор и анализ предметной области	25.02.2020 г.	
Разработка и реализация проекта	18.03.2020 г.	
Настройки работы веб-сервиса	21.04.2020 г.	

Подписи

консультантов и нормоконтролера на законченную дипломную работу (проект) с указанием относящихся к ним разделов работы (проекта)

Наименование разделов	Консультанты, Ф.И.О. (уч.степень, звание)	Дата подписания	Подпись
“Разработка онлайн-гардероба”	Иманбаев А. Ж.		
Нормоконтролер	Бауыржан М. Б.		
Программная часть	Кабдуллин М. А.		

Научный руководитель: _____ Иманбаев А. Ж.

Задание принял к исполнению обучающийся _____ Ахмет Е. Э.

Дата " ____ " _____ 20__ г.

Протокол анализа Отчета подобия Научным руководителем

Заявляю, что я ознакомился(-ась) с Полным отчетом подобия, который был сгенерирован Системой выявления и предотвращения плагиата в отношении работы:

Автор: Ахмет Еркебулан

Название: Разработка онлайн гардероба

Координатор: Азамат Иманбаев

Коэффициент подобия 1: 1

Коэффициент подобия 2: 0,9

Замена букв: 0

Интервалы: 0

Микропробелы: 0

Белые знаки: 0

После анализа Отчета подобия констатирую следующее:

- обнаруженные в работе заимствования являются добросовестными и не обладают признаками плагиата. В связи с чем, признаю работу самостоятельной и допускаю ее к защите;
- обнаруженные в работе заимствования не обладают признаками плагиата, но их чрезмерное количество вызывает сомнения в отношении ценности работы по существу и отсутствием самостоятельности ее автора. В связи с чем, работа должна быть вновь отредактирована с целью ограничения заимствований;
- обнаруженные в работе заимствования являются недобросовестными и обладают признаками плагиата, или в ней содержатся преднамеренные искажения текста, указывающие на попытки сокрытия недобросовестных заимствований. В связи с чем, не допускаю работу к защите.

Обоснование:

.....

.....
Дата

.....
Подпись Научного руководителя

Протокол анализа Отчета подобия

заведующего кафедрой / начальника структурного подразделения

Заведующий кафедрой / начальник структурного подразделения заявляет, что ознакомился(-ась) с Полным отчетом подобия, который был сгенерирован Системой выявления и предотвращения плагиата в отношении работы:

Автор: Ахмет Еркебулан

Название: Разработка онлайн гардероба

Координатор: Азамат Иманбаев

Коэффициент подобия 1:1

Коэффициент подобия 2:0,9

Замена букв:0

Интервалы:0

Микропробелы:0

Белые знаки:0

После анализа отчета подобия заведующий кафедрой / начальник структурного подразделения констатирует следующее:

- обнаруженные в работе заимствования являются добросовестными и не обладают признаками плагиата. В связи с чем, работа признается самостоятельной и допускается к защите;
- обнаруженные в работе заимствования не обладают признаками плагиата, но их чрезмерное количество вызывает сомнения в отношении ценности работы по существу и отсутствием самостоятельности ее автора. В связи с чем, работа должна быть вновь отредактирована с целью ограничения заимствований;
- обнаруженные в работе заимствования являются недобросовестными и обладают признаками плагиата, или в ней содержатся преднамеренные искажения текста, указывающие на попытки сокрытия недобросовестных заимствований. В связи с чем, работа не допускается к защите.

Обоснование:

.....
.....
.....
.....
.....
.....
.....

Дата

Подпись заведующего кафедрой /

начальника структурного подразделения

DOCUMENT.REPORT.SIMILARITY.TITLE:
Разработка онлайн гардероба
DOCUMENT.REPORT.SIMILARITY.AUTHOR:
Аюмет Еркебулан
DOCUMENT.REPORT.SIMILARITY.ORGANISATIONAL-UNIT:
ИКНИТ
DOCUMENT.REPORT.SIMILARITY.NUMBER-OF-CHECKS:
1
DOCUMENT.REPORT.SIMILARITY.DOCUMENT-SKIP-ADDRESSES:

DOCUMENT.REPORT.SIMILARITY.REPORT-DATE: 2020-05-24 17:44:19

DOCUMENT.REPORT.SIMILARITY.COORDINATOR:
Азамат Иманбаев
DOCUMENT.REPORT.SIMILARITY.UPLOAD-DATE:
2020-05-24 17:42:48

document.report.similarity.record-of-similarities

document.report.similarity.record.of.similarities.info

document.report.similarity.words 10541 document.report.similarity.words 3243 document.report.similarity.citations 1189

document.report.similarity.wp1-info document.report.similarity.wp2-info document.report.similarity.citations-coefficient

25 document.report.similarity.client-wc 3243 document.report.similarity.length-in-words 25699 document.report.similarity.length-in-chars

document.report.similarity.warnings-and-alerts

document.report.similarity.warnings-and-alerts.info

document.report.similarity.alert	0	in-the-text
document.report.similarity.spreads	0	in-the-text
document.report.similarity.micro-spaces	0	in-the-text
document.report.similarity.white-characters	0	in-the-text

document.report.similarity.active-lists-of-similarities

document.report.similarity.active.lists.info

document.report.similarity.top-ten-fragments (1,05 %)

DOCUMENT.REPORT.SIMILARITY.LP	DOCUMENT.REPORT.SIMILARITY.TITLE-OR-SOURCE	DOCUMENT.REPORT.SIMILARITY.AUTHOR	DOCUMENT.REPORT.SIMILARITY.WORDS.COPIED2
1	https://works.doklad.ru/view/_JMbGwM3BuM.html		28 0,86 %
2	https://works.doklad.ru/view/_JMbGwM3BuM.html		6 0,19 %

document.report.similarity.from-ref-books-database (0,00 %)

DOCUMENT.REPORT.SIMILARITY.LP	DOCUMENT.REPORT.SIMILARITY.TITLE	DOCUMENT.REPORT.SIMILARITY.AUTHOR	DOCUMENT.REPORT.SIMILARITY.WORDS.COPIED
DOCUMENT.REPORT.SIMILARITY.NO.SOURCES.FOUND			

document.report.similarity.from-the-home-base (0,00 %)

DOCUMENT.REPORT.SIMILARITY.LP	DOCUMENT.REPORT.SIMILARITY.TITLE	DOCUMENT.REPORT.SIMILARITY.AUTHOR	DOCUMENT.REPORT.SIMILARITY.DATE	DOCUMENT.REPORT.SIMILARITY.WORDS-FRAGMENTS
DOCUMENT.REPORT.SIMILARITY.NO.SOURCES.FOUND				

document.report.similarity.from-the-base-exchange-program (0,00 %)

DOCUMENT.REPORT.SIMILARITY.LP	DOCUMENT.REPORT.SIMILARITY.TITLE	DOCUMENT.REPORT.SIMILARITY.AUTHOR	DOCUMENT.REPORT.SIMILARITY.DATE	DOCUMENT.REPORT.SIMILARITY.WORDS-FRAGMENTS
DOCUMENT.REPORT.SIMILARITY.NO.SOURCES.FOUND				

document.report.similarity.from-the-internet (1,05 %)

DOCUMENT.REPORT.SIMILARITY.LP	DOCUMENT.REPORT.SIMILARITY.SOURCE-URL	DOCUMENT.REPORT.SIMILARITY.IDENTICAL-WORDS-FRAGMENTS
1	https://works.doklad.ru/view/_JMbGwM3BuM.html	34 (2) 1,05 %

АҢДАТПА

Бұл дипломдық жұмыста гардеробтағы операциялардың есебін онлайн режимінде жүргізуге, қолданушыларға төрт таңбалы код түрінде қағаз нөмірлерін бере отырып, веб-қосымшаны құру сипатталды.

Жобаның өзектілігі клиенттерге қызмет көрсету жылдамдығы мен дәлдігінің артуымен, нөмірлермен чиптердің жоғалуын жоюмен және вирустың таралу қаупі алдында қауіпсіздіктің жоғарылауымен байланысты.

Веб-бағдарлама MySQL және PHP, HTML, JavaScript (jQuery) бағдарламалау тілдерін қолдану арқылы жасалды. Диссертация жазу кезінде тақырыптық салаға талдау жасалды: интернеттегі гардеробты жобалау, енгізу және тестілеу.

АННОТАЦИЯ

В данной дипломной работе было описано создание веб-приложения, которое позволяет перевести учет операций в гардеробе в режим онлайн, выдавая пользователям бумажные номерки в виде четырехзначного кода.

Актуальность проекта обусловлена повышением скорости и точности обслуживания посетителей, устранением потерь фишек с номерами, повышением безопасности в условиях риска распространения вирусов.

Веб-приложение было создано с помощью MySQL и языков программирования PHP, HTML, JavaScript (jQuery). Во время написания дипломной работы был проведен анализ предметной области: проектирование, реализация и тестирование онлайн гардероба.

SUMMARY

This graduate work describes the creation of a web application, which allows to transfer the accounting of operations in the wardrobe to online, giving users paper numbers in the form of a four-digit code.

The relevance of the project is due to an increase in the speed and accuracy of customer service, elimination of the loss of chips with numbers, and increased security in the face of the risk of virus spread.

The web application was created using the MySQL and the programming languages PHP, HTML, JavaScript (jQuery). While writing a graduate work, an analysis of the subject area was carried out: design, implementation and testing of an online wardrobe.

СОДЕРЖАНИЕ

	ВВЕДЕНИЕ.....	9
1	ПОСТАНОВКА ЗАДАЧ ДИПЛОМНОЙ РАБОТЫ.....	11
2	РАЗРАБОТКА ОНЛАЙН ГАРДЕРОБА.....	14
2.1	Моделирование UML.....	14
2.2	Проектирование базы данных.....	16
2.3	Проектирование структуры информационной системы.....	18
2.4	Проектирование дерева функций и дерева диалога.....	20
2.5	Разработка программного кода.....	21
2.6	Тестирование онлайн гардероба.....	23
	ЗАКЛЮЧЕНИЕ.....	33
	СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ.....	35
	ПРИЛОЖЕНИЕ А.....	36
	ПРИЛОЖЕНИЕ Б.....	38

ВВЕДЕНИЕ

В настоящее время деятельность каждого предприятия и его структурного подразделения связана с непрерывным и большим потоком информации различного рода. По мере ускорения темпа жизни общества сотрудники организаций вынуждены приспосабливаться к растущему числу заявок, документов, данных для обработки. Причем время такой обработки тоже сокращается. Все это приводит к тому, что без применения средств автоматизации современные предприятия и учреждения не могут эффективно работать, ведь информатизация и цифровизация проникает во все сферы хозяйственной деятельности человека.

Для любой сферы деятельности важно получить максимальную отдачу от имеющихся ресурсов. Не является исключением в этой связи и деятельность гардероба, например в театре или любой другой организации, которая принимает одежду у посетителей.

Автоматизация – это средство для повышения эффективности процессов, используя современные технологии обработки информации. Причем эта эффективность не обязательно будет выражаться в стоимостных показателях. Это может быть повышение удобства, скорости работы, снижение числа ошибок сотрудников, снижение фактора потери или порчи инвентаря.

Автоматизация в общем случае позволяет достичь таких преимуществ по сравнению с не автоматизированными процессами:

- более эффективное распределение рабочей силы;
- снижение затрат;
- уменьшение человеческих ошибок;
- улучшение взаимодействия между сотрудниками;
- более глубокое понимание процессов, протекающих на предприятии.

Тема выпускной работы актуальна, поскольку автоматизация деятельности гардероба позволит повысить скорость и точность обслуживания посетителей, а также позволит избежать потерь карточек (фишек) с номерами, которые обычно выдаются взамен принятых вещей. Перевод процесса учета сданных вещей позволит выполнять анализ работы гардероба, повышая эффективность его деятельности. Очень важной является проблема распространения вирусных и бактериальных инфекций через многоразовые пластиковые или металлические номерки. Применение электронного онлайн-гардероба с одноразовыми бумажными номерками на ленте позволит повысить безопасность работников гардероба и посетителей, убрав дополнительный возможных канал распространения инфекций.

Целью выпускной работы является разработка автоматизированной информационной системы «Онлайн гардероб».

Объектом исследования является типичный гардероб, в котором отсутствуют какие-либо средства автоматизации его деятельности.

Предметом исследования является процесс проектирования и тестирования автоматизированной системы.

Для достижения поставленной цели требуется решить такие задачи:

- выполнить анализ предметной области;
- осуществить постановку задачи и выбор средств проектирования;
- выполнить проектирование, реализацию и тестирование онлайн гардероба.

1 ПОСТАНОВКА ЗАДАЧ ДИПЛОМНОЙ РАБОТЫ

Постановку задач удобно выполнить в виде технического задания. Техническое задание на веб-сервис «Онлайн Гардероб».

1.3.1 Общие сведения

1.3.1.1 Наименование программы

Автоматизированная информационная система «Онлайн Гардероб».

1.3.1.2 Краткая характеристика области применения

Это инструмент для автоматизированного управления процесса приема и выдачи вещей в типовом гардеробе.

1.3.2 Назначение разработки

1.3.2.1 Цели создания программы

Целью разработки веб-сервиса является снижение долгосрочных затрат отделов-гардеробов на закупку номерков, а также повышение удобства в процессе их деятельности, перевод информационных потоков в электронный вид. Также должна снизиться степень ответственности гардеробщиков, ведь они уже не будут отвечать за потерянные номерки и вообще не беспокоится об этой проблеме.

Очень важным в условиях рисков распространения вирусных инфекций является минимизация контактов с посетителями. При использовании многоразовых пластиковых или металлических номерков повышается риск передачи вирусов или бактерий через них. Предприятие вынуждено либо дезинфицировать их, тратя время и деньги, либо отказаться от их использования, перейдя на электронную систему.

1.3.2.2 Функциональное назначение программы

Разрабатываемая система предназначена для автоматизации процессов регистрации приема и выдачи вещей в типовом гардеробе.

Гардероб представляет собой последовательность стоек с вешалками (ячейками), которые выстроены в ряды.

Каждому гардеробщику сопоставлены определенные ряды. Рядом с рабочим местом гардеробщиков устанавливаются недорогие принтеры для ленточной бумаги. При приеме вещей принтер распечатывает автоматически сгенерированный код. После этого гардеробщик передает номерок клиенту.

При выдаче вещи гардеробщик вводит код с бумажного номерка в систему, и веб-сервис корректирует запись в базе данных.

1.3.3 Требования к программе или программному изделию

1.3.3.1 Требования к функциональным характеристикам

1.3.3.1.1 Требования к составу выполняемых функций

Веб-сервис должен выполнять такие функции:

- вести учет пользователей (гардеробщиков);
- позволять менять число номерного фонда в гардеробе;
- регистрировать прием и выдачу вещей;
- предусмотреть контроль ошибок и неправильных параметров.

1.3.3.1.2 Требования к организации входных данных

Сбор данных в системе должен осуществляться посредством ввода в поля параметров.

Входными данными программы являются следующие данные:

- параметры пользователей;
- четырехзначный код ячейки;
- номер ячейки.

1.3.3.1.3 Требования к организации выходных данных

Выходными данными будут являться:

- автоматически сгенерированный код, сопоставляемый ячейкам хранения;
- файл на печать на сетевой принтер.

1.3.3.2 Файловая структура

Приложение должно быть разработано в рамках стандартных правил и техник проектирования веб-ориентированных систем с клиент-серверной архитектурой.

1.3.3.3 Требования к оформлению макетов пользовательского интерфейса

Макеты пользовательского интерфейса должны быть логичны и понятны пользователю. Они должны быть разработаны на современных системах поддержки проектирования пользовательских интерфейсов.

1.3.3.4 Требования к надежности

Технология построения и работы приложения должна обеспечивать бесперебойный доступ пользователей к программе.

Данные требования должны быть реализованы посредством надежного процесса хранения, обработки и отображения данных.

1.3.3.5 Условия эксплуатации

Эксплуатация системы не должна требовать наличия специализированных знаний и навыков работы с ЭВМ.

Минимальные требования к серверу и клиенту представлены в таблицах 1.1 и 1.2.

Таблица 1.1 – Минимальные требования к серверу

Минимальные требования к серверу
Dual-Core Pentium CPU
2 GB DDR2RAM
250 GB SATA 7200 об./мин. HDD
Стандартный ИБП

Таблица 1.2 – Минимальные требования к клиенту

Минимальные требования к клиенту
Dual-Core CPU или эквивалент
2GB DDR RAM
250 GB HDD дискового пространства с 20% свободного места
Fast Ethernet NIC (поддержка протоколов TCP/IP)

1.3.3.6 Требования к программному обеспечению

Программное обеспечение должно:

- обладать функциональной полнотой для выполнения всех функций системы (п. 1.3.3.1.1. настоящего ТЗ);
- обладать модульностью построения;
- иметь стандартизованные информационные связи и использовать стандартные интерфейсы;
- допускать расширение функциональных возможностей системы;
- система должна быть открытой и распределенной;
- администрирование выполняется только в одном месте;
- обновление программного обеспечения должно производиться однократно на сервере, а не на каждой рабочей станции;
- программное обеспечение не должно быть чувствительно к программной и аппаратной платформе;
- наглядный, интуитивный интерфейс.

1.3.3.7 Требования к исходным кодам и языкам программирования

Программное обеспечение должно быть написано на языке программирования PHP для серверной части и HTML с JavaScript (jQuery) для клиентской части. Графическим фреймворком должен быть Bootstrap. В качестве СУБД следует использовать MySQL.

2 РАЗРАБОТКА ОНЛАЙН ГАРДЕРОБА

2.1 Моделирование UML

Моделирование – это проектирование программного приложения перед его кодированием на языке программирования. Унифицированный язык моделирования UML позволяет задавать, визуализировать и документировать модели программного обеспечения системы, включая их структуру и дизайн.

UML 2.0 определяет тринадцать типов диаграмм, разделенных на три категории – шесть типов диаграмм представляют статическую структуру приложения, три представляют собой общие типы поведения, и четыре представляют собой различные аспекты взаимодействия:

- структурные схемы включают в себя диаграмму классов, схему объектов, компонентную диаграмму, составную структурную диаграмму, диаграмму пакета, и диаграмму раскрытия;

- диаграммы поведения включают диаграмму вариантов использования (используется некоторыми методологии во время сбора требований), диаграммы деятельности, состояний;

- диаграммы взаимодействия включает диаграмму последовательности действий, диаграмму связи, диаграмму коммуникаций и временную диаграмму.

В рамках проектирования онлайн гардероба составим два типа диаграмм:

- диаграмма вариантов использования;

- диаграмма развертывания.

Диаграмма вариантов использования представлена на рисунке 2.1.

Выделяют три типа пользователей:

- неавторизованный пользователь;

- администратор;

- гардеробщик.

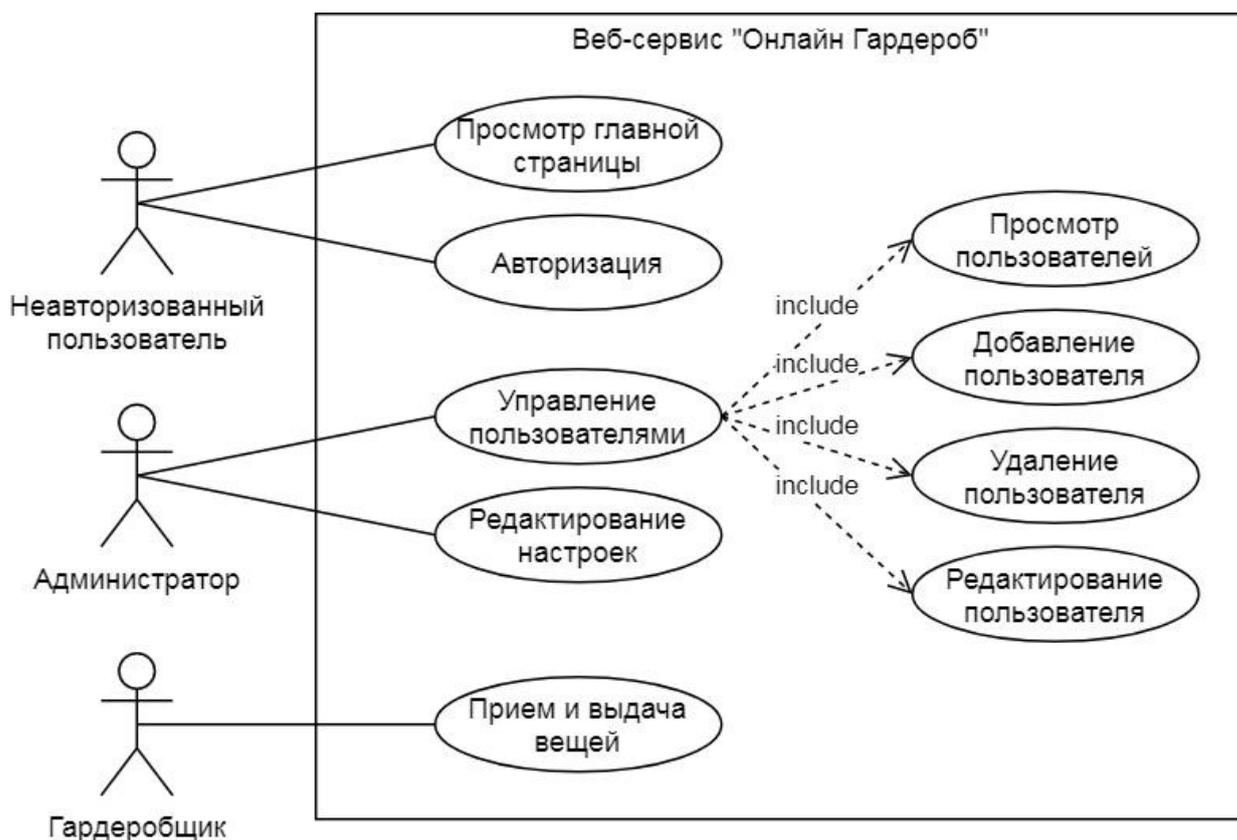


Рисунок 2.1 – Диаграмма вариантов использования

Неавторизованному пользователю доступен просмотр главной страницы, а также формы авторизации.

Администратор может выполнять управление пользователями (просмотр, добавление, удаление, редактирование), а также редактирование настроек системы.

Гардеробщик может регистрировать прием и выдачу вещей.

Диаграмма развертывания показана на рисунке 2.2.

Архитектура веб-сервиса представляет собой трехзвенное клиент-серверное приложение, в котором первым звеном является рабочая станция пользователей, вторым – сервер со скриптами приложения, а третьим – СУБД.

От множества клиентов поступают запросы на сервер, который их обрабатывает, запрашивая данные из базы данных или записывая их туда.

Также следует отметить, что при работе веб-сервиса скрипты обмениваются информацией и посылают на печать на сетевой принтер данные кода ячейки хранения.

Сетевой принтер реализует интерфейс (сокет), доступный по IP адресу и номеру порта.

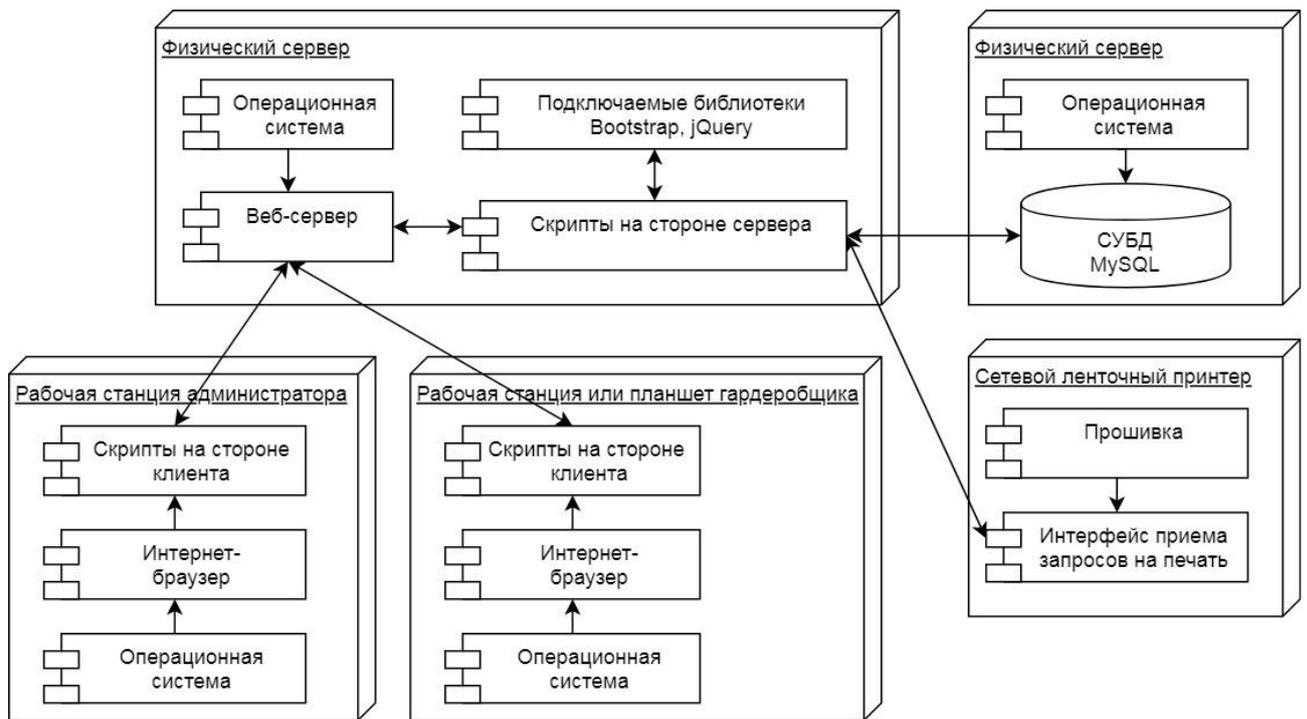


Рисунок 2.2 – Диаграмма развертывания

2.2 Проектирование базы данных

Проектирование базы данных включает четыре этапа:

- создание логической модели;
- создание физической модели;
- генерация кода SQL;
- импорт SQL-кода в СУБД.

При проектировании баз данных удобно пользоваться специализированным программным обеспечением, например, ERWin Data Modeler. В нем можно создать логическую и физическую модель, а также сгенерировать SQL код и даже экспортировать его в СУБД.

Логическая модель БД в ERWin представлена на рисунке 2.3.



Рисунок 2.3 – Логическая модель данных в ERWin

Три таблицы не связаны между собой. Но это нормально, так как при разработке данной базы в связывании нет необходимости.

Затем надо перейти к физической модели (рис.2.4).



Рисунок 2.4 – Переход к физической модели

Физическая модель БД представлена на рисунке 2.5.

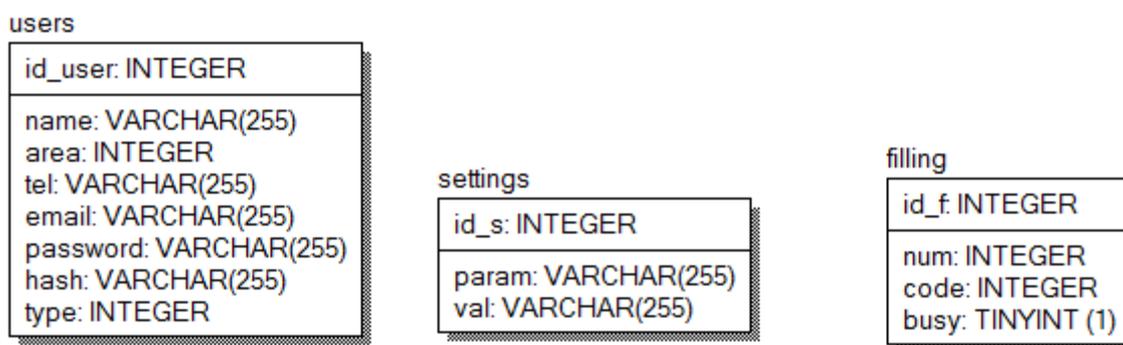


Рисунок 2.5 – Физическая модель БД

После этого надо воспользоваться инструментом Schema Generation для генерации SQL-кода (рис.2.6).

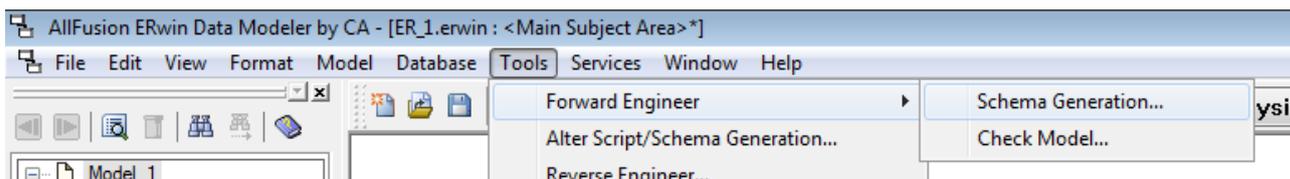


Рисунок 2.6 – Инструмент Schema Generation для генерации SQL кода

Затем следует создать пустую БД в phpmyadmin (рис.2.7).

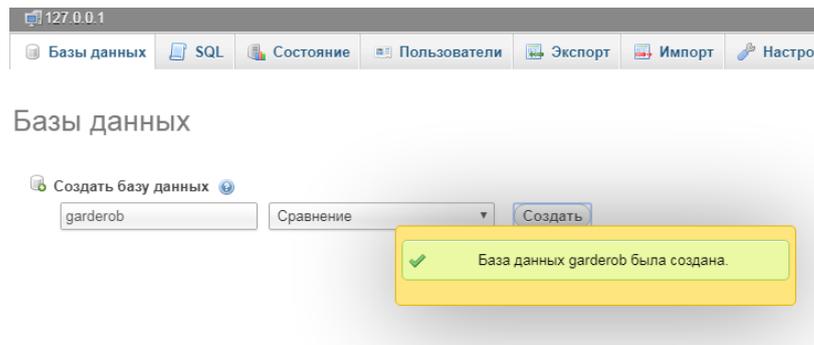


Рисунок 2.7 – Создание БД в СУБД

После импорта SQL-кода можно в дизайнера посмотреть получившуюся базу данных для онлайн гардероба (рис.2.8).

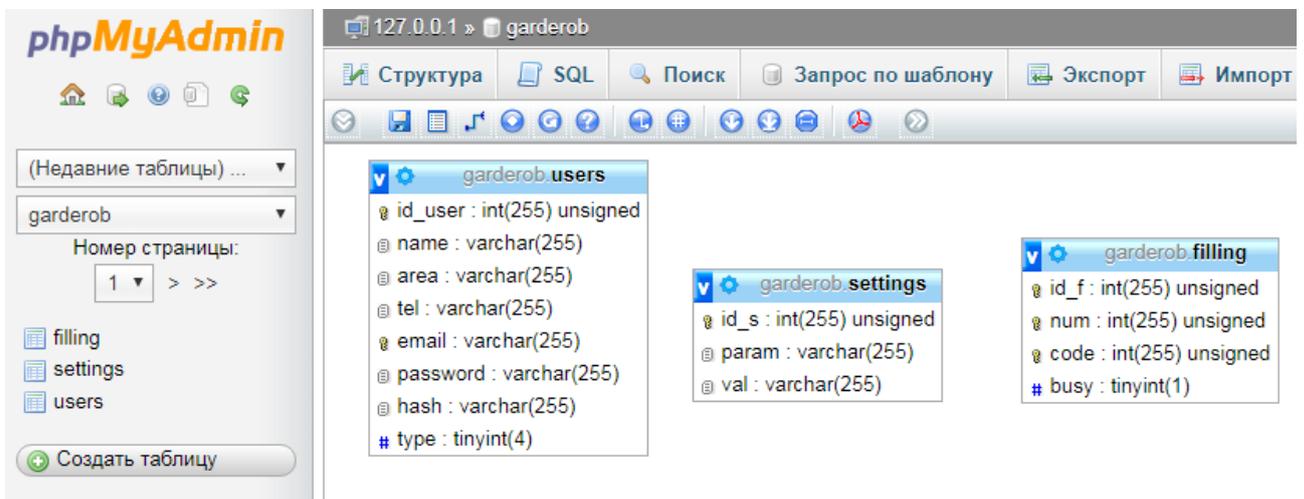


Рисунок 2.8 – Созданная БД в MySQL

SQL-код создания БД приведен в приложении А.

2.3 Проектирование структуры информационной системы

Структурная схема показывает состав модулей системы, а также выделяет их взаимосвязь (рис.2.9).

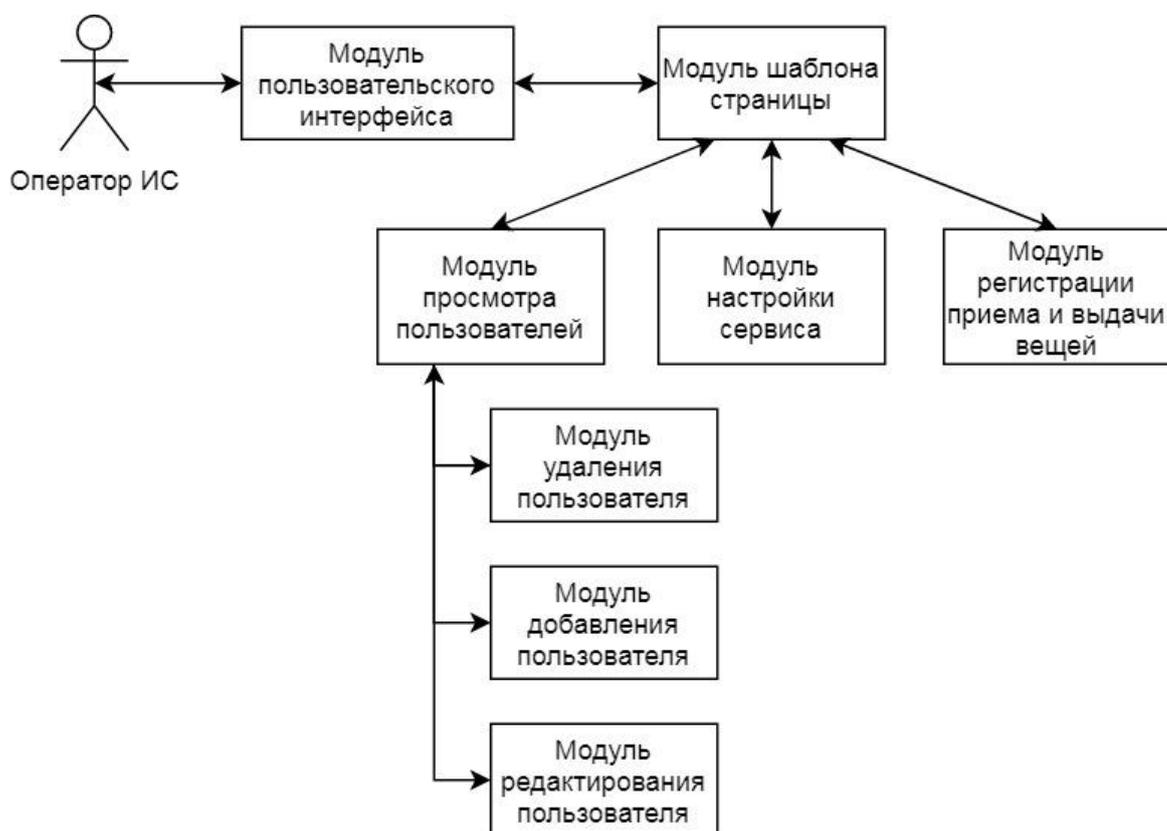


Рисунок 2.9 – Структура веб-системы

Оператор взаимодействует с модулем пользовательского интерфейса, который уже обращается к модулю шаблона страницы, подключающей модуль просмотра пользователей или модуль настройки сервиса или модуль регистрации приема и выдачи вещей.

Модуль просмотра пользователей вызывает модули создания, удаления, добавления пользователей ИС.

Связь модулей с файлами и библиотеками представлено в таблице 2.1.

Таблица 2.1 – Связь модулей с файлами и библиотеками

Название модуля	Файлы и библиотеки
Модуль пользовательского интерфейса	Bootstrap.js
Модуль шаблона страницы	index.php
Модуль просмотра пользователей	userList.php
Модуль удаления пользователя	userDelete.php
Модуль добавления пользователя	userAdd.php
Модуль редактирования пользователя	userEdit.php
Модуль настройки сервиса	settings.php
Модуль регистрации приема и выдачи вещей	garderob.php

2.4 Проектирование дерева функций и дерева диалога

Дерево функций показывает иерархию функций ИС (рис.2.10).

Выделяют такие функции:

а) основные;

- 1) регистрация приема и выдачи вещей;
- 2) вывод на печать кодов на бумажную ленту;

б) обеспечивающие:

- 1) управление пользователями;
- 2) настройки;
- 3) реализация пользовательского интерфейса;
- 4) ведение базы данных.



Рисунок 2.10 – Дерево функций

Дерево диалога показывает иерархию вызова экранных форм ИС и позволяет более продуманно проектировать алгоритмы и подчиненность при вызове страниц веб-сервиса (рис.2.11).

Вначале отрывается главная страница, которая носит чисто информативный характер. С этой страницы доступен вызов формы авторизации, из которой имеется доступ либо в личный кабинет администратора, либо в личный кабинет гардеробщика.



Рисунок 2.11 – Дерево диалога

В личном кабинете администратора можно изменять настройки сайта, а также управлять пользователями.

В личном кабинете гардеробщика можно регистрировать прием и выдачу вещей.

2.5 Разработка программного кода

В данном подразделе покажем лишь некоторые примеры программного кода спроектированного веб-сервиса онлайн гардероба.

Основные настройки сайта принято выносить в отдельный файл `config.php`. На рисунке 2.12 представлена структура файла `config.php`:

```

<?php
    define('DATABASE_NAME','garderob');
    define('DATABASE_USER','root');
    define('DATABASE_PASSWORD','');
    define('DATABASE_HOST','localhost');
    define('SERVER','http://localhost');
    define('PREFIX','/garderob.online/');
    define('SITE_NAME','Онлайн гардероб');
    define('SITE_DEBUG', false);
    define('REAL_WORK', false);
    // Настройки принтера кодов
    define('PRINTER_IP', '192.168.1.33');
    define('PRINTER_PORT', 9100);
?>

```

Рисунок 2.12 – Листинг файла config.php

Здесь находятся параметры подключения к базе данных, настройки имени и адреса сайта.

Параметр 'SITE_DEBUG' включает отображение ошибок, а также отображает элементы массивов \$_POST, \$_GET, что удобно при программировании обновляемых форм.

Параметр 'REAL_WORK' задает режим работы приложения. Если он установлен в TRUE, то система при каждом приеме вещи будет посылать данные на печать на сетевой принтер с адресом 'PRINTER_IP' и портом 'PRINTER_PORT'.

На рисунке 2.13 представлен код отправки данных на печать:

```

// Отправка на печать кода ячейки
if(REAL_WORK){
    try{
        $fp=pfsockopen(PRINTER_IP, PRINTER_PORT);
        fputs($fp, date("Y-m-d H:i:s")." \r\n". $code);
        fclose($fp); }
    catch (Exception $e){
        if(SITE_DEBUG){
            echo "<p>&nbsp;</p><div class='alert alert-danger'>";
            echo 'Ошибка: ', $e->getMessage(), "\n";
            echo "</div>"; }}}

```

Рисунок 2.13 – Фрагмент кода отправки данных на печать

На печать выводится текущая дата и время и код. Эти данные на листе (части бумажной ленты) передаются клиенту, отдавшему вещи.

Так как при тестировании на локальной машине сетевой принтер не подключался, то это приводило к большим задержкам при приеме заявок, ведь скрипт пытался около 30 секунд подключиться к удаленному принтеру.

На рисунке 2.14 представлены маршрутизация и запрещение страниц в модуле шаблона страницы:

```
if(!$SESSION['uID']) // неавторизованный пользователь – запрещение страниц
    if(in_array($url, array("settings", "garderob", "userLogout", "userDelete", "userEdit", "userList", "userAdd")))
        redir2('404');
if($SESSION['uID'] && !$SESSION['admin']) // авторизованный, но не администратор – запрещение страниц
    if(in_array($url, array("settings", "userDelete", "userEdit", "userList", "userAdd")))
        redir2('404');
if($SESSION['uID'] && $SESSION['admin']) // авторизованный, администратор – запрещение страниц
    if(in_array($url, array("garderob")))
        redir2('404');
```

Рисунок 2.14 – Листинг маршрутизации и запрещения страниц в модуле шаблона страницы

Здесь контролируется доступ групп пользователей к модулям системы и блокируется, если прав недостаточно.

Листинги программных модулей приложения описаны в приложении Б.

2.6 Тестирование онлайн гардероба

Тестирование веб-сервиса производилось на локальном хостинге под управлением веб-сервера Apache.

Для доступа к сервису следует в строке запросов браузера ввести «<http://localhost/garderob.online>».

После этого будет загружена и отображена заглавная страница (рис.2.15). Она не несет никакой информации и служит для дальнейшего перехода на форму авторизации.

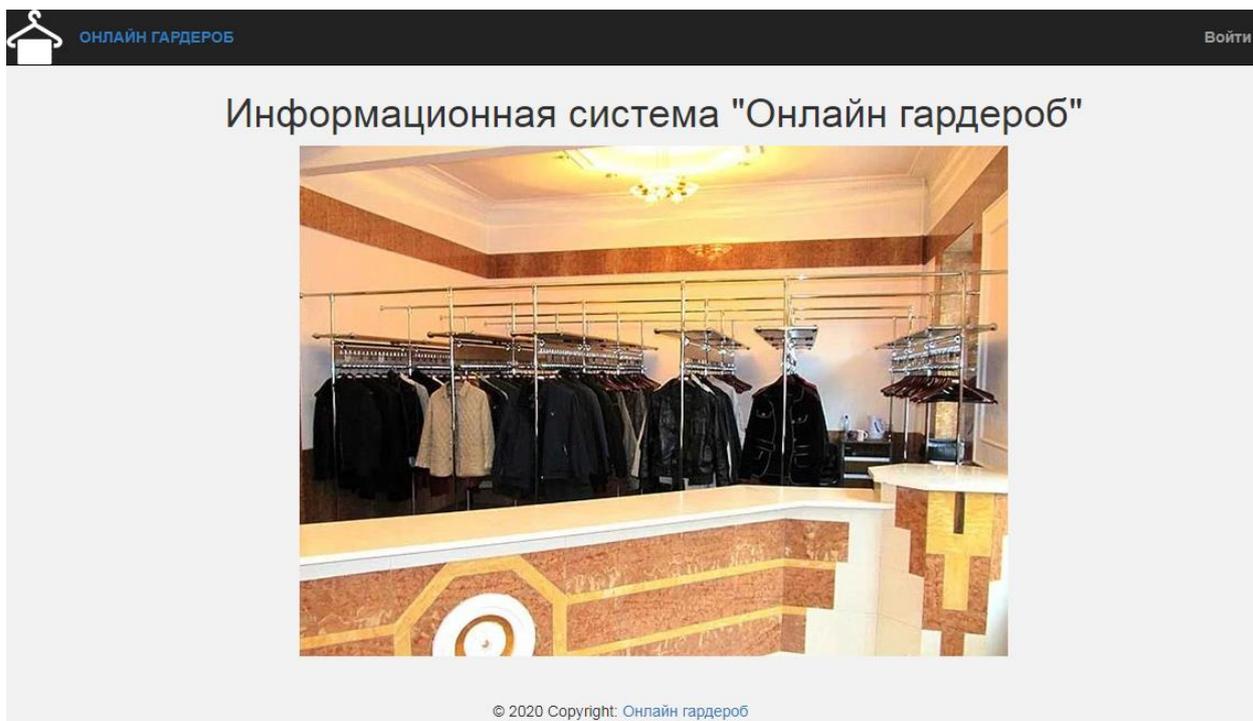


Рисунок 2.15 – Главная страница веб-сервиса

Выполним авторизацию под аккаунтом администратора (рис.2.16).

Рисунок 2.16 – Форма авторизации пользователей

После авторизации отображается личный кабинет администратора, где сразу он может приступить к редактированию пользователей (рис.2.17).

Надо понимать, что в общем случае гардероб представляет собой ряды с вешалками или с ячейками, которым присвоены номера по порядку. Поэтому для задания общего количества ячеек следует указать число зон хранения (ячеек) в ряду, а также количество рядов.

Обычно в гардеробе присутствует несколько работников, каждый из которых отвечает за свои определенные ряды. Поэтому в настройках гардеробщиков есть поле «Ряды ответственности».

Панель администратора: управление пользователями

[Добавить новую запись](#)

ФИО	Ряды ответственности	Телефон	Email	Пароль	Тип	Действия
Администратор		+7 (727) 435454	admin@garderoob.online	***	Администратор ИС	редактировать
Исаева Динара	1,2	+7 (727) 5656546	babicheva@garderoob.online	***	Гардеробщик	редактировать удалить
Алиева Айгуль	3,4	+7 (727) 5656443	salkova@garderoob.online	***	Гардеробщик	редактировать удалить
Садыкова Дария	5,6	+7 (727) 46754533	lopatina@garderoob.online	***	Гардеробщик	редактировать удалить
Абдуллаева Малика	7,8	+7 (727) 2344354	komarova@garderoob.online	***	Гардеробщик	редактировать удалить

© 2020 Copyright: [Онлайн гардероб](#)

Рисунок 2.17 – Форма просмотра пользователей

При клике на ссылку «редактировать» открывается форма редактирования пользователя (рис.2.18).

Редактирование пользователя

Введите имя обязательное поле

Введите ряды ответственности (1,2,5) обязательное поле

Введите email обязательное поле

Введите пароль обязательное поле

Введите телефон не обязательное поле

Тип пользователя

Кнопка подтверждения [СОХРАНИТЬ ИЗМЕНЕНИЯ](#)

Рисунок 2.18 – Форма редактирования пользователя

Гардеробы бывают разные по числу рядов и количеству ячеек в ряду, поэтому предусмотрена форма изменения данных параметров (рис.2.19).

Рисунок 2.19 – Форма настроек веб-приложения

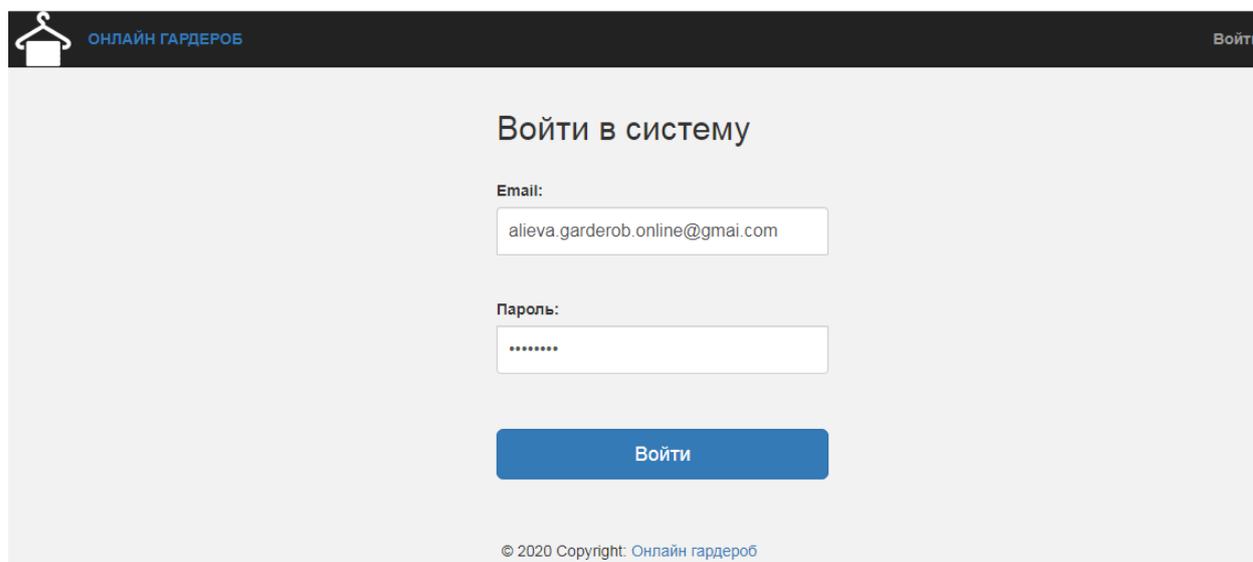
Перед изменением выводится предупреждение, потому что произойдет перезапись и сброс таблицы хранения (рис.2.20).

Рисунок 2.20 – Предупреждение о перезаписи базы данных

После изменения выводится сообщение (рис.2.21).

Рисунок 2.21 – Сообщение об успешном обновлении БД

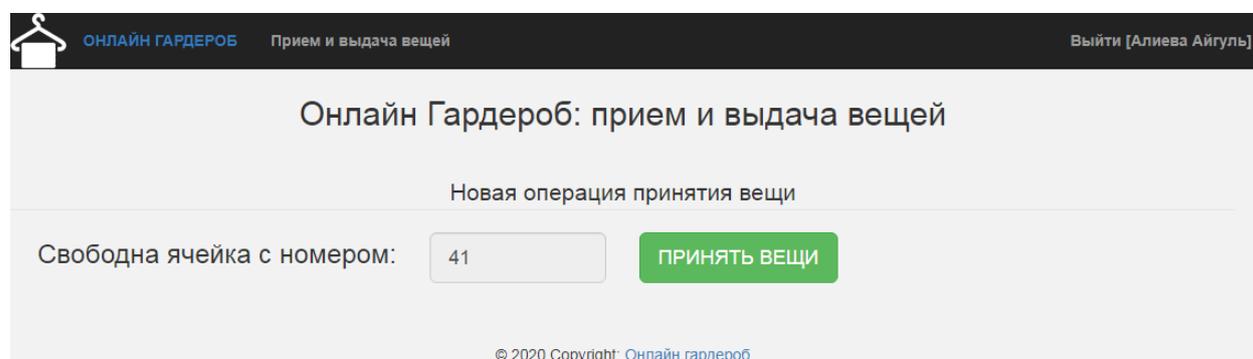
Теперь покажем функционал, доступный гардеробщику. Для этого надо авторизоваться под его аккаунтом (рис.2.22).



The screenshot shows the login interface for 'Онлайн Гардероб'. At the top left is a logo with a hanger icon and the text 'ОНЛАЙН ГАРДЕРОБ'. At the top right is a 'Войти' button. The main heading is 'Войти в систему'. Below it are two input fields: 'Email:' with the value 'alieva.garderoob.online@gmail.com' and 'Пароль:' with masked characters '*****'. A blue 'Войти' button is positioned below the password field. At the bottom, there is a copyright notice: '© 2020 Copyright: Онлайн гардероб'.

Рисунок 2.22 – Авторизация под аккаунтом гардеробщика

После авторизации сразу отображается форма приема и выдачи вещей (рис.2.23).



The screenshot shows the 'Онлайн Гардероб: прием и выдача вещей' page. The top navigation bar includes the logo, 'ОНЛАЙН ГАРДЕРОБ', 'Прием и выдача вещей', and a 'Выйти [Алиева Айгуль]' button. The main heading is 'Онлайн Гардероб: прием и выдача вещей'. Below it is the sub-heading 'Новая операция принятия вещи'. The form contains the text 'Свободна ячейка с номером:' followed by an input field containing the number '41' and a green button labeled 'ПРИНЯТЬ ВЕЩИ'. At the bottom, there is a copyright notice: '© 2020 Copyright: Онлайн гардероб'.

Рисунок 2.23 – Личный кабинет гардеробщика

Поскольку база данных пустая, то есть еще не принимались вещи, то показана только подформа принятия.

После нажатия на кнопку «Принять вещи» будет зарегистрирована операция, сгенерирован код (в данном случае 8533), который будет отправлен на печать на сетевой ленточный (бумага, свернутая в ленту) принтер.

Четырехзначный код быстро вводить, и достаточно низкая вероятность, что кто-то специально будет говорить не свой код, чтобы забрать чужие вещи.

Этот код будет показан на экране (рис.2.24). Также код будет распечатан на ленте вместе с датой и временем операции. Это аналог номерка. Гардеробщик берет из принтера этот номерок на бумаге и выдает его клиенту.

ОНЛАЙН ГАРДЕРОБ Прием и выдача вещей Выйти [Алиева Айгуль]

Онлайн Гардероб: прием и выдача вещей

Принято в ячейку 41. Код: **8533**

Новая операция принятия вещи

Свободна ячейка с номером: **ПРИНЯТЬ ВЕЩИ**

Новая операция выдачи вещи

Введите четырехзначный код:
или
Введите номер ячейки:

ВЫДАТЬ ВЕЩИ

0 1 2 3 4 5 6 7 8 9 **ОЧИСТИТЬ**

Рисунок 2.24 – Сообщение о принятии вещи и сгенерированный код

Преимуществом является то, что даже если пользователь потеряет бумажный номерок, это не приведет к потерям для предприятия. Не нужно закупать новые номерки, ведь они печатаются на бумаге, которая стоит недорого и все равно выбрасывается после использования.

Если клиент вернулся и говорит, что потерял номер, то гардеробщик в поле выдачи вещи может ввести номер ячейки, а не присвоенный ей динамически код. Но большинство клиентов, как показывает практика, все-таки номерки не теряет, поэтому это редкие случаи выдачи вещей по номеру ячейки.

Принтер автоматически обрезает ленту бумаги, так что гардеробщик просто достает уже готовый бумажный номерок из принтера.

То есть гардеробщику просто надо нажимать на кнопки принятия вещей (рис.2.25).

ОНЛАЙН ГАРДЕРОБ Прием и выдача вещей Выйти [Алиева Айгуль]

Онлайн Гардероб: прием и выдача вещей

Принято в ячейку 43. Код: **9244**

Новая операция принятия вещи

Свободна ячейка с номером: **ПРИНЯТЬ ВЕЩИ**

Новая операция выдачи вещи

Введите четырехзначный код:
или
Введите номер ячейки:

ВЫДАТЬ ВЕЩИ

0 1 2 3 4 5 6 7 8 9 **ОЧИСТИТЬ**

Рисунок 2.25 – Принятие последующей вещи в свободную ячейку

Для выдачи вещей надо воспользоваться кнопкой «Выдать вещи». При этом надо ввести или четырехзначный код, или номер ячейки (если клиент потеряет бумажный номерок) (рис.2.26).

ОНЛАЙН ГАРДЕРОБ Прием и выдача вещей Выйти [Алиева Айгуль]

Онлайн Гардероб: прием и выдача вещей

Выдача из ячейки 41 по коду **8533**

Новая операция принятия вещи

Свободна ячейка с номером: **ПРИНЯТЬ ВЕЩИ**

Новая операция выдачи вещи

Введите четырехзначный код:
или
Введите номер ячейки:

ВЫДАТЬ ВЕЩИ

0 1 2 3 4 5 6 7 8 9 **ОЧИСТИТЬ**

Рисунок 2.26 – Выдача вещи по ранее сгенерированному коду

Если ввести не существующий номер в базе или номер пустой ячейки, то система выдаст предупреждение (рис.2.27).

The screenshot shows the 'Онлайн Гардероб' interface. At the top, there is a navigation bar with a home icon, the text 'ОНЛАЙН ГАРДЕРОБ Прием и выдача вещей', and a user profile 'Выйти [Алиева Айгуль]'. The main heading is 'Онлайн Гардероб: прием и выдача вещей'. A prominent red message box states: 'Ячейка с кодом "2212" не найдена'. Below this, there are two sections for operations. The first section, 'Новая операция принятия вещи', shows 'Свободна ячейка с номером:' with an input field containing '41' and a green button labeled 'ПРИНЯТЬ ВЕЩИ'. The second section, 'Новая операция выдачи вещи', shows 'Введите четырехзначный код: или Введите номер ячейки:' with an empty input field and a blue button labeled 'ВЫДАТЬ ВЕЩИ'. At the bottom, there is a numeric keypad with buttons for digits 0-9 and a blue button labeled 'ОЧИСТИТЬ'.

Рисунок 2.27 – Выдача по коду, которого не существует

Если все ячейки заняты, то выдается предупреждение (рис.2.28).

The screenshot shows the 'Онлайн Гардероб' interface. At the top, there is a navigation bar with a home icon, the text 'ОНЛАЙН ГАРДЕРОБ Прием и выдача вещей', and a user profile 'Выйти [Алиева Айгуль]'. The main heading is 'Онлайн Гардероб: прием и выдача вещей'. A green message box states: 'Принято в ячейку 60. Код: 5485'. Below this, a yellow warning box states: 'Свободных мест не осталось. Надо сначала выдать вещи'. The interface then shows the 'Новая операция выдачи вещи' section with 'Введите четырехзначный код: или Введите номер ячейки:' and an empty input field, followed by a blue button labeled 'ВЫДАТЬ ВЕЩИ'. At the bottom, there is a numeric keypad with buttons for digits 0-9 and a blue button labeled 'ОЧИСТИТЬ'.

Рисунок 2.28 – Обработка ситуации заполнения всех ячеек

После выдачи ячейки по номеру 43 (клиент потерял номерок, поэтому ввод идет по номеру ячейки) система вновь позволяет ее заполнить (рис.2.29).

Онлайн Гардероб: прием и выдача вещей

Выдача по номеру ячейки 43

Новая операция принятия вещи

Свободна ячейка с номером: 43 **ПРИНЯТЬ ВЕЩИ**

Новая операция выдачи вещи

Введите четырехзначный код:
или
Введите номер ячейки: **ВЫДАТЬ ВЕЩИ**

0 1 2 3 4 5 6 7 8 9 **ОЧИСТИТЬ**

Рисунок 2.29 – Выдача по коду 6492

Также система позволяет запускаться на смартфоне или планшете, потому что его верстка адаптивная (рис.2.30).

Внизу формы представлена онлайн клавиатура, что позволяет эффективно пользоваться приложением с тачскрином. Для ввода четырехзначного кода в систему теперь не надо ставить фокус на поле ввода и ждать загрузки встроенной в ОС клавиатуры, можно просто нажимать по этим электронным кнопкам на странице.

А подсистема принятия вещей вообще работает без ввода цифр, автоматически выбирая ближайшую свободную ячейку.

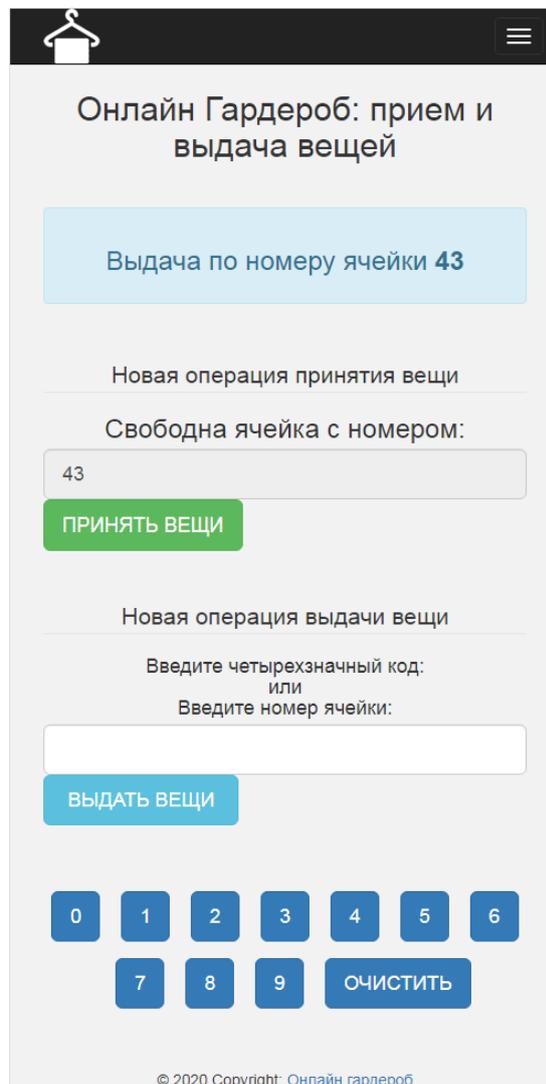


Рисунок 2.30 – Просмотр в режиме смартфона

Таким образом, разработанный веб-сервис позволяет перевести учет операций в гардеробе в режим онлайн, выдавая пользователям бумажные номерки. Бумага для таких принтеров (например, кассовая) стоит очень недорого, принтера также недорогие, зато на длительную перспективу гардероб избавлен от проблемы потерянных номерков.

Также в условиях угрозы распространения вирусных инфекций применение одноразовых бумажных номерков повышает уровень безопасности гардеробщиков и посетителей.

ЗАКЛЮЧЕНИЕ

В данной выпускной работе выполнено проектирование и тестирование веб-ориентированной системы «Онлайн Гардероб».

Вначале выполнена постановка задачи.

Целью разработки веб-сервиса являлось снижение долгосрочных затрат отделов-гардеробов на закупку номерков, а также повышение удобства в процессе их деятельности, перевод информационных потоков в электронный вид. Также должна снизиться степень ответственности гардеробщиков, ведь они уже не будут отвечать за потерянные номерки и вообще не беспокоится об этой проблеме.

Каждому гардеробщику сопоставлены определенные ряды. Рядом с рабочим местом гардеробщиков устанавливаются недорогие принтеры для ленточной бумаги. При приеме вещей принтер распечатывает автоматически сгенерированный код. После этого гардеробщик передает номерок клиенту.

В качестве средств проектирования выбраны:

- языки программирования PHP, JavaScript (jQuery), HTML;
- СУБД MySQL;
- графический фреймворк Bootstrap.

Проектирование системы начато с разработки UML диаграмм:

- диаграммы вариантов использования;
- диаграммы развертывания.

Проектирование базы данных проходило в четыре этапа:

- создание логической модели;
- создание физической модели;
- генерация кода SQL;
- импорт SQL-кода в СУБД.

Структурная схема показала состав модулей системы, а также выделило их взаимосвязь.

Дерево функций показало иерархию функций ИС, а дерево диалога – иерархию вызова экранных форм ИС и позволило более продуманно спроектировать алгоритмы и подчиненность при вызове страниц веб-сервиса.

Тестирование веб-сервиса «Онлайн Гардероб» разработанный веб-сервис позволяет перевести учет операций в гардеробе в режим онлайн, выдавая пользователям бумажные номерки. Бумага для таких принтеров (например, кассовая) стоит очень недорого, принтера также недорогие, зато на длительную перспективу гардероб избавлен от проблемы потерянных номерков.

Также в условиях угрозы распространения вирусных инфекций применение одноразовых бумажных номерков повышает уровень безопасности гардеробщиков и посетителей.

Применение электронного онлайн-гардероба с одноразовыми бумажными номерками на ленте позволит повысить безопасность работников гардероба и посетителей, убрав дополнительный возможных канал распространения инфекций.

Онлайн клавиатура системы позволяет эффективно пользоваться приложением с тачскрином. Для ввода четырехзначного кода в систему теперь не надо ставить фокус на поле ввода и ждать загрузки встроенной в ОС клавиатуры, можно просто нажимать по этим электронным кнопкам на странице. А подсистема принятия вещей вообще работает без ввода цифр, автоматически выбирая ближайшую свободную ячейку.

Таким образом, в рамках выпускной работы решены такие задачи:

- выполнен анализ предметной области;
- осуществлена постановка задач и выбор средств проектирования;
- выполнено проектирование, реализация и тестирование онлайн гардероба.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

- 1 Емельянова Н. Проектирование информационных систем. – М.: Форум, 2014. – 432 с.
- 2 Затонский А. В. Программирование и основы алгоритмизации / А.В. Затонский, Н.В. Бильфельд. – М.: Дрофа, 2014. – 176 с.
- 3 Исаев Г. Проектирование информационных систем. – М.: «Омега-Л», 2012. – 432 с.
- 4 Козленко Л. Проектирование информационных систем // КомпьютерПресс. – № 9. – 2013. – С. 3-17.
- 5 Маклафлин Б. PHP и MySQL. Исчерпывающее руководство. – Изд. Питер, 2016. – 544 с.
- 6 Мандел Т. Дизайн интерфейсов, ДМК, 2015 – 410 с.
- 7 Род С. Алгоритмы. Теория и практическое применение. – М.: «Эксмо», 2017. – 544 с.
- 8 Советов Б.Я. Архитектура информационных систем: учебник для студ. учреждений высш. проф. образования. – М.: Издательский центр «Академия», 2012. – 288 с.
- 9 Тарасов С. СУБД для программиста. Базы данных изнутри. – Омск: «Соломон», 2015. – 320 с.

ПРИЛОЖЕНИЕ А

```
CREATE TABLE IF NOT EXISTS users(  
    id_user int(255) unsigned NOT NULL AUTO_INCREMENT,  
    name varchar(255) NOT NULL,  
    area varchar(255) NOT NULL DEFAULT "",  
    tel varchar(255) DEFAULT NULL,  
    email varchar(255) NOT NULL,  
    password varchar(255) NOT NULL,  
    hash varchar(255) NOT NULL,  
    type tinyint(4) NOT NULL DEFAULT '2',  
    PRIMARY KEY(id_user),  
    UNIQUE KEY email(email)  
) ENGINE = InnoDB DEFAULT CHARSET = utf8 AUTO_INCREMENT = 6;  
  
INSERT INTO users(id_user, name, area, tel, email, password, hash, type) VALUES  
    (1, 'Администратор', '', '+7 (727) 435454',  
'admin.garderob.online@gmail.com', 'QDhuBpWE', '', 1),  
    (2, 'Исаева Динара', '1', '+7 (727) 5656546',  
'isaeva.garderob.online@gmail.com', 'u5A1lKkA', '', 2),  
    (3, 'Алиева Айгуль', '2', '+7 (727) 5656443',  
'alieva.garderob.online@gmail.com', 'ABDfJWoO', '', 2),  
    (4, 'Садыкова Дария', '3', '+7 (727) 46754533',  
'sadykova.garderob.online@gmail.com', 'PZgM4XEE', '', 2),  
    (5, 'Абдуллаева Малика', '4', '+7 (727) 2344354',  
'abdullaeva.garderob.online@gmail.com', 'dDSbNtu4', '', 2);  
  
CREATE TABLE IF NOT EXISTS settings(  
    id_s int(255) unsigned NOT NULL AUTO_INCREMENT,  
    param varchar(255) NULL,  
    val varchar(255) NULL,  
    PRIMARY KEY(id_s)  
)  
ENGINE=InnoDB DEFAULT CHARSET = utf8;  
  
INSERT INTO settings(param, val) values  
    ('num_in_row', 1),  
    ('rows_cnt', 4);  
  
CREATE TABLE IF NOT EXISTS filling(  
    id_f int(255) unsigned NOT NULL AUTO_INCREMENT,  
    num int(255) unsigned NULL,  
    code int(255) unsigned NULL,  
    busy tinyint(1) NOT NULL DEFAULT 0,
```

ПРОДОЛЖЕНИЕ ПРИЛОЖЕНИЯ А

```
PRIMARY KEY(id_f),  
UNIQUE (num),  
UNIQUE (code)  
)  
ENGINE = InnoDB DEFAULT CHARSET = utf8;  
  
INSERT INTO filling(num) VALUES (1), (2), (3), (4);
```

ПРИЛОЖЕНИЕ Б

Листинг Б.1 – Модуль настройки приложения:

```
<h1>Настройки веб-приложения</h1>
<?php
    if(isset($_POST['settings']))
    {
        $NumInRow = trim($_POST['NumInRow']);
        $RowsCnt = trim($_POST['RowsCnt']);

        $sql1 = "INSERT INTO filling(num) values ";
        $cnt = $NumInRow*$RowsCnt;
        $i=1;
        for(; $i<$cnt; $i++)
            $sql1 .= "($i),";
        $sql1 .= "($i),";

        $sql = "UPDATE settings set val=$NumInRow WHERE param like
'num_in_row'";
        sqlOther($sql);
        $sql = "UPDATE settings set val=$RowsCnt WHERE param like
'rows_cnt'";
        sqlOther($sql);
        $sql = "DROP TABLE IF EXISTS filling";
        sqlOther($sql);
        $sql = "CREATE TABLE IF NOT EXISTS filling(
id_f int(255) unsigned NOT NULL AUTO_INCREMENT,
num int(255) unsigned NULL,
code int(255) unsigned NULL,
busy tinyint(1) NOT NULL DEFAULT 0,
PRIMARY KEY (id_f),
UNIQUE (num),
UNIQUE (code)
) ENGINE = InnoDB DEFAULT CHARSET = utf8";
        sqlOther($sql);
        sqlOther($sql1);

        echo "<div class='alert alert-success'>Обновление базы данных
успешно завершено</div>";
    }
    else
    {
        $sql = "select val from settings where param like 'num_in_row'";
```

ПРОДОЛЖЕНИЕ ПРИЛОЖЕНИЯ Б

```
$NumInRow = sqlSelect($sql);
$NumInRow = $NumInRow[0][0];
$sql = "select val from settings where param like 'rows_cnt';";
$RowsCnt = sqlSelect($sql);
$RowsCnt = $RowsCnt[0][0];
}
?>
<form class="form-horizontal" onsubmit="if(!window.confirm('При изменении
настроек текущее состояние гардероба будет сброшено. Продолжить?')) return
false;" method="post" enctype="multipart/form-data" accept-charset="utf-8">
<fieldset>
<legend></legend>
<div class="form-group">
<label class="col-md-3 control-label" for="NumInRow">Число зон хранения в
ряде</label>
<div class="col-md-3">
<input id="NumInRow" name="NumInRow" type="number" placeholder=""
class="form-control input-md" required="" value='<?php if
(isset($_POST['settings'])) echo $_POST['NumInRow']; else if (isset($NumInRow)
and $NumInRow>0) echo $NumInRow; ?>'>
<span class="help-block">обязательное поле</span>
</div>
<label class="col-md-3 control-label" for="RowsCnt">Число рядов</label>
<div class="col-md-3">
<input id="RowsCnt" name="RowsCnt" type="number" placeholder="" class="form-
control input-md" required="" value='<?php if (isset($_POST['settings'])) echo
$_POST['RowsCnt']; else if (isset($RowsCnt) and $RowsCnt>0) echo $RowsCnt;
?>'>
<span class="help-block">обязательное поле</span>
</div>
</div>
<div class="form-group">
<label class="col-md-4 control-label" for="settings">Кнопка
подтверждения</label>
<div class="col-md-4">
<button id="settings" type="submit" name="settings" class="btn btn-
primary">СОХРАНИТЬ ИЗМЕНЕНИЯ</button>
</div>
</div>
</fieldset>
</form>
```

ПРОДОЛЖЕНИЕ ПРИЛОЖЕНИЯ Б

Листинг Б.2 – Модуль регистрации приема и выдачи вещей:

```
<h2>Онлайн Гардероб: прием и выдача вещей</h2>
<?php
    if($_SESSION['admin'])
    {
        echo "<p>&nbsp;</p><div class='alert alert-danger'>Администратор
не может принимать и выдавать вещи</div>";
        exit();
    }

    // Получение настроек
    $sql = "select val from settings where param like 'num_in_row'";
    $NumInRow = sqlSelect($sql);
    $NumInRow = $NumInRow[0][0];
    $sql = "select val from settings where param like 'rows_cnt'";
    $RowsCnt = sqlSelect($sql);
    $RowsCnt = $RowsCnt[0][0];

    // Получение позволенных рядов
    $sql = "select area from users where id_user=".$_SESSION['uID'];
    $area = sqlSelect($sql);
    $area = explode(",", $area[0][0]);

    if(isset($_POST['fill'])) // Принятие вещей
    {
        $num = $_POST['freeCell'];
        while(sqlSelect("select num from filling where code=".(generateHash($code =
generateHash($code)))));
        updateTable("filling", "busy, code", array(1, $code), "num", $num);
        echo "<p>&nbsp;</p><div class='alert alert-success'><h3>Принято в
ячейку $num. Код: <b>$code</b></h3></div>";

        // Отправка на печать кода
        if(REAL_WORK)
        {
            try
            {
                $fp=pfsockopen(PRINTER_IP, PRINTER_PORT);
                fputs($fp, date("Y-m-d H:i:s")." \r\n". $code);
                fclose($fp);
            }
        }
    }
}
```

ПРОДОЛЖЕНИЕ ПРИЛОЖЕНИЯ Б

```
catch (Exception $e)
{
    if(SITE_DEBUG)
    {
        echo "<p>&nbsp;</p><div class='alert alert-
danger'>";
        echo 'Ошибка: ', $e->getMessage(), "\n";
        echo "</div>";
    }
}

if(isset($_POST['take'])) // Выдача вещей
{
    $code = trim($_POST['code']);
    if(strlen($code)==2) // Номер ячейки
    {
        $sql = "select num from filling where num=$code and busy=1 and
(";
        foreach($area as $RowNum)
        {
            $from = $RowNum*$NumInRow;
            $to = $from + $NumInRow;
            $from++;
            $sql .= "(num between $from and $to) OR ";
        }
        $sql = substr($sql, 0, -4);
        if(substr($sql, $len-4, $len) == " OR ")
            $sql = substr($sql, 0, $len-4);
        $sql .= ")";
        $num = sqlSelect($sql);
        if(!$num)
        {
            echo "<p>&nbsp;</p><div class='alert alert-
danger'><h3>Заполненная ячейка с номером \"<b>$code</b>\" не
найден</h3></div>";
        }
        else
        {
            $num = $num[0][0];
        }
    }
}
```

ПРОДОЛЖЕНИЕ ПРИЛОЖЕНИЯ Б

```
updateTable("filling", "busy, code", array(0, 'null'), "num",
$code);
echo "<p>&nbsp;</p><div class='alert alert-
info'><h3>Выдача из ячейки $num по номеру ячейки
<b>$code</b></h3></div>";
}
}
else // Код ячейки
{
    $sql = "select num from filling where code=$code and busy=1";
    $num = sqlSelect($sql);
    if(!$num)
    {
        echo "<p>&nbsp;</p><div class='alert alert-
danger'><h3>Ячейка с кодом \"<b>$code</b>\" не найдена</h3></div>";
    }
    else
    {
        $num = $num[0][0];
        updateTable("filling", "busy, code", array(0, 'null'), "code",
$code);
        echo "<p>&nbsp;</p><div class='alert alert-
info'><h3>Выдача из ячейки $num по коду <b>$code</b></h3></div>";
    }
}
}

// Получение ближайшей свободной ячейки. Проверка занята ли хотя бы
одна ячейка
$sql = "select id_f, num, busy from filling where busy=0 and (";
$sql1 = "select count(num) from filling where busy=1 and (";
foreach($area as $RowNum)
{
    $from = $RowNum*$NumInRow;
    $to = $from + $NumInRow;
    $from++;
    $sql .= "(num between $from and $to) OR ";
    $sql1 .= "(num between $from and $to) OR ";
}
$len = strlen($sql);
if(substr($sql, $len-4, $len) == " OR ")
```

ПРОДОЛЖЕНИЕ ПРИЛОЖЕНИЯ Б

```
$sql = substr($sql, 0, $len-4);
$len1 = strlen($sql1);
if(substr($sql1, $len1-4, $len1) == " OR ")
    $sql1 = substr($sql1, 0, $len1-4);
$sql .= ") limit 1";
$sql1 .= "));";
$freeCell = sqlSelect($sql);
if(!$freeCell)
    echo "<div class='alert alert-warning'><h3>Свободных мест не
осталось. Надо сначала выдать вещи</h3></div>";
$busyCnt = sqlSelect($sql1);
$busyCnt = $busyCnt[0][0];
?>
<p>&nbsp;</p>
<form class="form-horizontal" method="post" enctype="multipart/form-data"
accept-charset="utf-8">
<fieldset>
<?php if($freeCell){ ?>
<legend>Новая операция принятия вещи</legend>
<div class="form-group">
<label class="col-md-4 control-label" for="freeCell"><h3
style="margin:0;">Свободна ячейка с номером:</h3></label>
<div class="col-md-2">
<input id="freeCell" name="freeCell" type="number" readonly placeholder=""
class="form-control input-lg" required="" value='<?php echo $freeCell[0][1]; ?>'>
</div>
<div class="col-md-2 text-left">
<button id="fill" type="submit" name="fill" class="btn btn-success btn-
lg">ПРИНЯТЬ ВЕЩИ</button>
</div>
</div>
<?php } ?>
<?php if($busyCnt>0){ ?>
<p>&nbsp;</p>
<legend>Новая операция выдачи вещи</legend>
<div class="form-group">
<label class="col-md-4 control-label" for="code" style="padding-top:0;"><h4
style="margin:0;padding:0;">Введите четырехзначный код:<br>или<br>Введите
номер ячейки:</h4></label>
<div class="col-md-2">
<input id="code" name="code" type="number" placeholder="" class="form-control
input-lg" value="">
```

