

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»**
(**Н И У « Б е л Г У »**)

ИНСТИТУТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ И ЕСТЕСТВЕННЫХ НАУК
КАФЕДРА МАТЕМАТИЧЕСКОГО И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
ИНФОРМАЦИОННЫХ СИСТЕМ

**АВТОМАТИЗИРОВАННАЯ СИСТЕМА ПЛАНИРОВАНИЯ
ЭКЗАМЕНА В ГИБДД УМВД**

Выпускная квалификационная работа
обучающегося по направлению подготовки
02.03.03 Математическое обеспечение и администрирование
информационных систем
очной формы обучения,
группы 07001302
Сивирин Маргариты Александровны

Научный руководитель
доцент
Румбешт В.В

БЕЛГОРОД 2017

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	4
1 ПОСТАНОВКА ЗАДАЧИ НА РАЗРАБОТКУ АС.....	6
1.1 Общие определения.....	6
1.2 Обзор и анализ систем, реализующих подобные задачи.....	9
1.2.1 Задачи и требования автоматизации системы.....	9
1.2.2 Многофункциональный центр предоставления государственных и муниципальных услуг (МФЦ).....	10
1.2.3 Одно окно.....	13
1.2.4 Электронная регистратура.....	13
1.2.5 Госуслуги.....	15
1.3 Требования к системе.....	17
1.3.1 Назначения и цели создания АСПЭ.....	18
1.3.2 Функциональные требования.....	19
1.3.3 Требования к системе.....	21
2 РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ.....	24
2.1 Проектирование структуры базы данных.....	24
2.2 Проектирование структуры WEB – приложения.....	28
2.3 Программная реализация.....	31
3 ИСПЫТАНИЕ.....	40
3.1 Програма и методика испытаний.....	40
3.2 Результаты испытаний.....	41
ЗАКЛЮЧЕНИЕ.....	47

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ	49
ПРИЛОЖЕНИЕ	51

ВВЕДЕНИЕ

В нашем информационном обществе существует множество автоматизированных информационных систем в различных сферах человеческой деятельности, связанных с предоставлением услуг, обработкой информации, предоставлением данных.

Несоответствие требованиям автоматизации определяет необходимость разработки автоматизированной системы.

Одной из функций ГИБДД УМВД является функция выдачи водительского удостоверения. При выполнении этой функции возникает проблема неравномерно распределенной очереди и неверный расчёт времени для сдачи экзамена для получения водительского удостоверения. Актуальность обусловлена потребностью автоматизации планирования экзамена, а именно очереди, которая улучшит процесс сдачи экзамена.

Целью выпускной квалификационной работы является более устойчивая автоматизация процесса сдачи экзамена и уменьшение затраченного времени.

Для достижения цели мы поставили перед собой ряд задач:

1. Обзор и анализ проведения экзамена в ГИБДД УМВД;
2. Обзор и анализ существующих систем, а именно:
 - 1) многофункционального центра предоставления государственных и муниципальных услуг;
 - 2) одно окно;
 - 3) электронной регистратуры;
 - 4) госуслуг.
3. Формулировка требований к системе;
4. Проектирование автоматизированной системы;

5. Программная реализация системы;
6. Проведение испытаний.

Успешными результатами работы автоматизированной системы планирования экзамена в ГАИ являются:

1. Уменьшение затраченного времени на подачу документов для сдачи экзамена в ГИБДД УМВД;
2. Уменьшение затраченного времени для сдачи и приема экзамена в ГИБДД УМВД;
3. Предоставление пользователю полного доступа к информационным материалам, связанным с актуальными вопросами сдачи(пересдачи) экзамена в ГИБДД УМВД;
4. Повышение информированности пользователя по актуальным вопросам сдачи экзамена;
5. Улучшение информационного взаимодействия между пользователем и сотрудниками ГИБДД;
6. Упрощение процесса записи на все этапы для сдачи экзамена в ГИБДД.

Использование данной АС позволит сделать процесс сдачи экзамена удобнее и быстрее как для экзаменуемых, так и для сотрудников подразделений ГИБДД.

Выпускная квалификационная работа содержит: 50 листов, 1 таблицу, 18 рисунков, 8 листингов.

1 ПОСТАНОВКА ЗАДАЧИ НА РАЗРАБОТКУ АС

1.1 Общие определения

Экзамен в ГИБДД УМВД – тестирование человека на наличие теоретических и практических умений и навыков управления транспортным средством, в результате которого человеку выдается водительское удостоверение.

Водительское удостоверение (в просторечии — «водительские права») — документ, удостоверяющий личность и подтверждающий право на управление соответствующими категориями транспортных средств.

Национальные водительские удостоверения соответствуют требованию «Конвенции о дорожном движении (Приложение б)», согласно которому удостоверения печатаются на языке или языках, установленных органами власти, которые выдают или которым поручено выдавать этот документ. В удостоверениях указываются название или отличительные знаки страны, в которой они выдаются (для России— знак «RUS»). Национальные водительские удостоверения признаются в странах, подписавших «Конвенцию о дорожном движении», то есть практически во всех странах Европы. Если же водители попадают в страны, не подписавшие эту «Конвенцию», то чтобы водить там автомобиль, водителю придётся сдавать экзамены для получения водительского удостоверения этой страны.

Категории транспортных средств, позволяющие управлять мототранспортом, автотранспортом, троллейбусом и трамваем (с 2014 года):

1) категории «А», «А1», «М» — мотоцикл, с рабочим объёмом двигателя от 50 до 125 куб. см и максимальной мощностью до 11 кВт; мопеды и скутеры, а также лёгкие квадроциклы с рабочим объёмом двигателя до 50 куб. см;

2) категории «В», «В1», «ВЕ» — автомобили, с разрешённой максимальной массой, не превышающей 3,5 т и число сидячих мест, кроме сиденья водителя, не более восьми; трициклы и квадрициклы; автомобили с прицепом, разрешённая максимальная масса которого превышает 750 кг;

3) категории «С», «С1», «СЕ», «С1Е» — автомобили, с разрешённой максимальной массой от 3,5 т до 7,5 т, в том числе с прицепом до 750 кг

4) категории «D», «D1», «DE», «D1E» — автомобили, предназначенные для перевозки пассажиров и имеющие больше 8 сидячих мест, кроме места водителя, в том числе с прицепом до 750 кг; оборудованные количеством сидячих мест от 9 до 16, кроме водительского; сцепленный с прицепом, с разрешённой максимальной массой, превышающей 750 кг, но не превышающей 3,5 т;

5) категории «Тм», «Тв» — трамваи; троллейбусы.

К сдаче экзамена на право управления самоходными машинами допускаются лица, достигшие возраста:

- для самоходных машин категории «А1» — 16 лет;
- для самоходных машин категорий «В», «С», «Е», «F» — 17 лет;
- для самоходных машин категории «D» — 18 лет.

Допуск к сдаче экзамена на получение водительского удостоверения в ГИБДД производится, если человек прошел обучение в организации, имеющей лицензию на подобное обучение (на категории «А» и «В» допускалась самостоятельная подготовка до 5 ноября 2013 года). С 2014 года с 16 лет можно открыть категории «А1» и «М»; категории «А», «В» («В1») и «С» («С1») с 18 лет; категория «D» («D1») доступна с 21 года (при условии открытой категории «С» («С1»)); для «прицепных» категорий «ВЕ», «СЕ» («С1Е») или «DE» («D1E») необходимо иметь стаж вождения тягача соответствующей категории не менее 1 года. Экзамен состоит из трех этапов: теория правил дорожного движения (ПДД), автодром(площадка) и город.

Тестирование теоретической части производится на компьютерах. На решение теоретической части отводится 20 минут. Тест состоит из 20

вопросов одного билета. Допускается сделать 2 ошибки. В отличие, от различных тренировочных программ, при сдаче экзамена в ГИБДД, у экзаменуемого нет возможности узнать верность ответа на каждый вопрос. Все свои ошибки и их количество экзаменуемый узнает только после ответа на все 20 вопросов билета. При положительном результате сдачи теоретической части далее следует второй этап – площадка.

Сотрудники ГИБДД назначают экзаменуемому три упражнения на свой выбор, из четырех, которые экзаменуемый отработывал во время обучения в специализированной школе, это «Симметричная змейка», «Остановка и трогание на подъеме», «Параллельная парковка задним ходом» или «Разворот». Сотрудник ГИБДД наблюдает за сдачей экзамена со стороны, т.е. во время сдачи экзамена в машине находится только экзаменуемый. В случае если и с этим этапом экзаменуемый справился – отправляется на заключительный этап – город.

Вожделение по городу — третий, заключительный этап экзамена. К нему допускаются только те экзаменуемые, которые успешно сдали первые два этапа, а именно: теорию ПДД и площадку. Несколько экзаменуемых садятся в автомобиль: один за руль, остальные — на пассажирские сиденья. Место рядом с водителем занимает сотрудник ГИБДД. Он сообщает экзаменуемым, по какому маршруту необходимо следовать. Список маршрутов составляется заранее, поэтому информация должна быть вывешена на стенде в отделении ГИБДД. Экзаменуемый проезжает определенный отрезок пути. За каждую ошибку сотрудник ГИБДД выставляет определенное количество баллов. Баллы суммируются. Если число штрафных баллов менее 5, экзаменуемый получает оценку «Сдал». При 5 и более баллах ставится оценка «Не сдал».

Первая пересдача разрешается по истечении 7 дней после неудачного прохождения этапа. Результаты теории и автодрома действительны в течении 3 месяцев. Если экзаменуемого не удалось сдать город в этот период времени, первые два этапа придется сдавать заново.

1.2 Обзор и анализ систем, реализующих подобные задачи

В данном разделе мы рассмотрим несколько существующих систем, выполняющих аналогичные задачи. Так же мы выдвинем ряд требований, по которым будем анализировать системы и составим сводную таблицу на соответствия требованиям.

1.2.1 Задачи и требования автоматизации системы

Автоматизированная система (АС) - это система, состоящая из персонала и комплекса средств автоматизации его деятельности, реализующая автоматизированную технологию выполнения установленных функций.

Целью автоматизации информационных процессов является повышение производительности и эффективности труда работников, улучшение качества информационной продукции и услуг, повышение сервиса и оперативности обслуживания пользователей.

Автоматизация базируется на использование средств вычислительной техники (СВТ) и необходимого программного обеспечения (ПО).

Основными задачами автоматизации процессов являются:

1. ускорение процессов обработки информации;
2. повышение оперативности и качественного уровня обслуживания пользователей;
3. облегчение возможностей обмена информацией;
4. модернизация элементов традиционных технологий.

Разновидностью АС являются информационные системы (ИС), основной целью которых является хранение, обеспечение эффективного поиска и передачи информации по соответствующим запросам.

Решением задач автоматизации являются требования:

1. узкая направленность;

2. легкость и удобность использования;
3. полнота и актуальность информации;
4. точность и достоверность информации.

Существует достаточно большое количество автоматизированных информационных систем (АИС), реализующих задачи по поиску, хранению и обработке информации. АИС предназначены для обеспечения людей информацией, а значит должны быть удобными в использовании; информация должна быть актуальной, полной и достоверной для удовлетворения конкретных информационных потребностей в рамках определенной предметной области, значит АИС должна быть узко направленной. Результатом работы автоматизированной информационной системы является информационный продукт: данные, документы, услуги.

1.2.2 Многофункциональный центр предоставления государственных и муниципальных услуг (МФЦ)

МФЦ - это единое место приема, регистрации и выдачи документов гражданам и организациям при предоставлении государственных и муниципальных услуг.

МФЦ обеспечит быстрое и комфортное получение услуг, даст возможность получить одновременно несколько взаимосвязанных услуг, предоставит достоверную и актуальную информацию, необходимую для получения услуг.

На панели главного меню веб-сайта МФЦ (рис. 1.1), пользователю предложена функция, которая предоставляет перечень всех выполняемых многофункциональным центром услуг. В полный перечень услуг входит около 100 видов работ, которые разделены на 7 групп:

1. Правопорядок и безопасность;
2. Социальная поддержка населения;
3. Операции с недвижимостью;

4. Определение или подтверждение гражданско-правового статуса;
5. Регулирование предпринимательской деятельности;
6. Сопутствующие и дополнительные услуги;
7. Услуги по жизненным ситуациям.

Каждая группа разделена в среднем на 2-4 подгруппы, которые, в свою очередь, разделены на 1-3 узконаправленные категории.

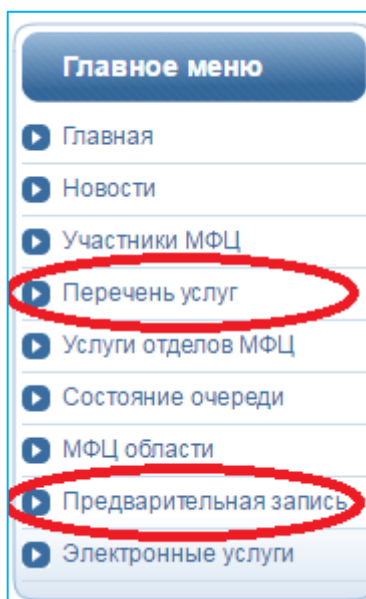


Рис. 1.1. Главное меню

Так же на главной панели меню, пользователю предложена услуга предварительной записи (см. рис. 1.1). При нажатии на выбранный пункт меню, на странице появляется информация о том, как происходит запись на прием, какими способами (по телефону или через интернет) можно осуществить запись и на какие услуги осуществляется запись только при личном обращении к администратору. Так же на странице отображается ссылка с названием «Предварительная запись через интернет»

Анализ программы проводился два раза с перерывом 10 дней.

На странице предварительной записи пользователю предоставляется выбор услуги. В перечне выбора имеется лишь одна выполняемая услуга «Услуги реестра».

Далее в графе узкой направленности услуги предложен выбор лишь двух направлений: «Выдача документов» и «Прием документов на кадастровый учет недвижимого имущества» (рис. 1.2).

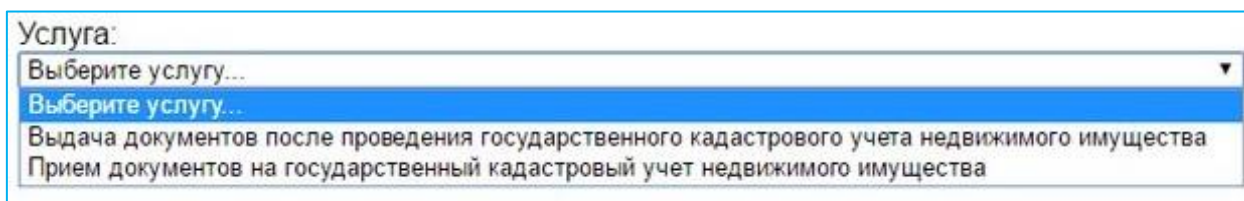


Рис. 1.2. Услуги

При анализе программы в первый раз, был замечен сбой, который дублирует направления услуг (рис. 1.3).

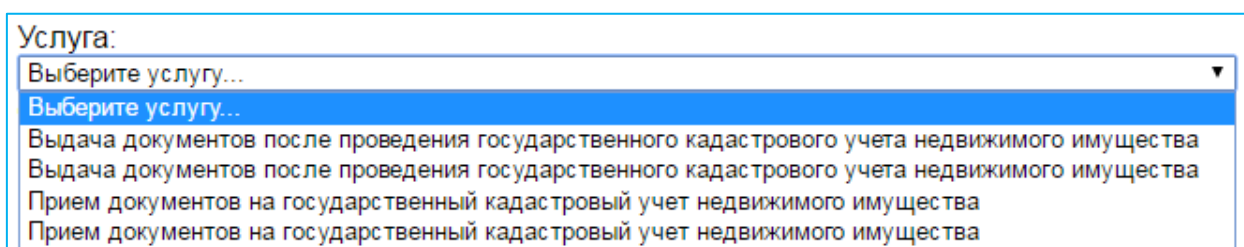


Рис. 1.3. Сбой в графе «Услуги»

Выбор даты предоставлен на неделю вперед, включая выходные дни: суббота и воскресенье. После выбора даты предложен выбор времени, после чего заполняются поля «Фамилия» и «Номер телефона» с подтверждением записи через смс.

Как уже было сказано выше, МФЦ - это место предоставления услуг, которое обеспечивает быстрое и комфортное получение услуг, предоставляет необходимую достоверную и актуальную информацию. Но, проанализировав программу, можно сделать вывод, что программа работает не так хорошо, как хотелось бы. Во-первых, МФЦ много направленная программа, и, возможно, это приводит к некоторым сбоям в работе, которые могут ввести пользователя в заблуждения. Во-вторых, электронная запись возможна только на 2 услуги из 100 имеющихся в многофункциональном центре, а значит программа не предоставляет полную информацию пользователю. Так

как актуальность и достоверность мы проверить не можем, будем предполагать, что этим критериям программа соответствует.

1.2.3 Одно окно

Одно окно – центр обслуживания граждан и юридических лиц по вопросам кадастрового учета и земельных работ.

На главной странице автоматизированной информационной системы «Одно окно» пользователю сразу предоставляется список из 13 выполняемых услуг. В главном меню так же имеется перечень услуг. При нажатии на любую из видов работ, на странице появляется более подробная информация о выбранной услуге. В системе отсутствует информация о процессе записи на прием, так же, как и электронная запись. Обновления новостей датируются 2010 годом, из чего можно предположить, что полной информации АИС не дает, актуальность не гарантирует, как и точность и достоверность информации.

1.2.4 Электронная регистратура

При входе на главную страницу электронной регистратуры, система предлагает пользователю выбор региона и населённого пункта. Без предыдущего сбора информации, система не дает возможность пользователю получить следующий набор данных. Например, пользователь не сможет выбрать медицинское учреждение, не выбрав предыдущем пунктом необходимый регион.

После выбора региона, пользователю предоставляется выбор медицинского учреждения. Кроме полного списка государственных медицинских учреждений, пользователю предоставляется фильтр, которой сортирует и группирует медицинские учреждения по критериям: «Все», «Взрослые», «Детские».

Далее следует выбор специализации узких специалистов. На экране пользователь может видеть профиль (кардиология, терапия) и количество сводных талонов по конкретному профилю. Так же на экране в виде закладок отображаются все этапы сбора информации от начального (выбор региона), до конечного (получение талона). Пользователь имеет возможность вернуться на нужную закладку, минуя предыдущие, одним кликом мыши.

После выбора специализации, пользователю предоставляется возможность выбора конкретного специалиста, даты и времени. В этом помогает информация о ФИО врача и кабинете, в котором он принимает, дате и времени приема. Если в определенный день есть свободный талон, то день окрашен зеленым, если нет – серым. В ходе сбора данных, программа формирует талон, в котором видны все предыдущие выбранные пункты пользователем (рис. 1.4).

Рис. 1.4. Формирование талона

Далее пользователю необходимо выбрать время приема. После выбора времени, система информирует об авторизации через АИС «Госуслуги», без которой пользователь не сможет осуществить запись.

Автоматизированная информационная система «Электронная регистратура» соответствует требованиям, которым должна соответствовать любая АИС, но она является ветвью более глобальной АИС «Госуслуги»,

поэтому следует заметить, что регистрация через побочную АИС не является удобной в использовании.

1.2.5 Госуслуги

«Госуслуги» представляют собой онлайн сервисы, содержащие информацию о государственных услугах для граждан и организаций. В данную АИС входит 16 разных категорий, таких как «Пенсии», «Работа», «Здоровье», «Лицензии», «Бизнес», «Образование», «Налоги».

Анализ автоматизированной информационной системы «Госуслуги» мы проводили в категории «Транспорт и вождение». В эту категорию входит 5 видов услуг:

1. Штрафы ГИБДД;
2. Регистрация транспортного средства;
3. Водительское удостоверение;
4. Разрешение на перевозку тяжелых грузов;
5. Информация из реестра автомобильных дорог.

Далее в каждой категории имеются 2-5 узких направлений. Мы тестировали направление «Получение водительского удостоверения после прохождения профессионального обучения на право управления транспортным средством».

После выбора нужной услуги, на странице АИС появилась информация о процессе получения выбранной услуги: авторизация, документы, заявление. После информирования пользователя, появляется форма для заполнения личной информации для подачи электронного заявления, которая разделена на несколько пунктов:

1. ФИО, дата рождения, номер телефона и адрес электронной почты;
2. Паспортные данные;
3. Адрес проживания;
4. Сведения об автошколе;

5. Сведения о медицинской справке.

Все поля формы обязательны к заполнению, в ином случае заявление не отправляется. При заполнении формы в пункте «Сведения о медицинской справке» возникли сложности. Форма требовала информацию о номере медицинской справки и номера лицензии медицинского учреждения (рис. 1.5).

6 Сведения о медицинской справке

Номер справки 8337	Дата выдачи 23.09.2010
Название медицинского учреждения поликлиника №7	Номер лицензии

Рис. 1.5. Сведения о медицинской справке

На моей справке номер отсутствовал, как и номер лицензии медицинского учреждения. После ввода придуманных мною цифр в графу номера справки и лицензии, заявление было отправлено и получило статус «Рассматривается». Спустя неделю пришло оповещение с информацией о приемных днях и времени для сдачи экзамена на получение водительского удостоверения.

Исходя из проделанного анализа, автоматизированная информационная система не настроена на проверку достоверности информации, а значит проверка документов происходит вручную сотрудниками ГАИ в день сдачи экзамена. Можно сделать вывод, что в заполнении нескольких пунктов формы для подачи электронного заявления нет необходимости. Следовательно, пользователь теряет удобность и легкость пользования информационной системой и тратит свое время.

Результаты анализа автоматизированных информационных систем на предмет наших требований приведены в таблице 1.1.

Таблица 1.1

Результаты анализа

	МФЦ	Одно окно	Электронная регистратура	Госуслуги
Узкая направленность АИС	-	+	+	-
Легкость и удобность использования АИС	+	+	-	-
Полнота информации	-	-	+	+
Актуальность информации	+	-	+	+
Точность и достоверность информации	+	-	+	+

Исходя из анализа видно, что анализируемые автоматизированные информационные системы имеют недостатки и не соответствуют хотя бы одному из поставленных нами критериев. При разработке нашей информационной системы, мы учитывали все указанные нами необходимые критерии.

1.3 Требования к системе

В данном разделе мы сформируем требования к нашей автоматизированной системе, укажем назначение и цели создания системы.

1.3.1 Назначения и цели создания АСПЭ

АСПЭ предназначена для решения задач создания единого информационного пространства, обеспечивающего возможности:

1. Предоставления пользователю полного доступа к информационным материалам, связанным с актуальными вопросами сдачи(пересдачи) экзамена в ГИБДД УМВД;

2. Уменьшения затраченного времени на подачу документов для сдачи экзамена в ГИБДД УМВД;

3. Уменьшение затраченного времени для сдачи и приема экзамена в ГИБДД УМВД.

К автоматизированной деятельности относятся:

1. Процессы сбора, получения, обработки и предоставления информации посредством АСПЭ;

2. Технические и организационные мероприятия по сопровождению АСПЭ.

В перечень объектов автоматизации входят:

1. Объекты и структурные подразделения ГИБДД УМВД;

2. Сервисы АСПЭ;

3. Технологические процессы, связанные с указанными сервисами.

Целями создания АСПЭ являются:

1. Повышение информированности пользователя по актуальным вопросам сдачи экзамена;

2. Улучшение информационного взаимодействия между пользователем и сотрудниками ГИБДД;

3. Уменьшение затраченного времени на сдачу экзамена пользователем, и принятия экзамена сотрудниками ГИБДД;

4. Упрощение процесса записи на все этапы для сдачи экзамена в ГИБДД.

1.3.2 Функциональные требования

АИС автоматизирует работу сотрудников ГИБДД и обеспечивает выполнение следующих функций:

1. регистрация пользователей;
2. выдача индивидуального номера идентификации на время сдачи экзамена;
3. запись экзаменуемого на удобное для него время и дату на каждый из этапов;
4. проверка количества попыток сдачи каждого из этапов;
5. информирование экзаменуемого о возможной дате пересдачи этапа;
6. информирование экзаменуемого о ходе экзамена;
7. построение диаграмм о количестве экзаменуемых прошедших с первой попытки, прошедших с нескольких попыток и не прошедших экзамен вообще за (промежуток времени);
8. хранение информации о пользователях (период времени).

Входные данные:

- ФИО;
- серия и номер паспорта экзаменуемого;
- выбор желаемой даты сдачи экзамена пользователем.

Выходные данные:

1. номер пользователя (присваивается системой автоматически);
2. количество попыток прохождения каждого из этапов;
3. дата прохождения следующего этапа (если пользователь осуществил запись);
4. отчет (список) экзаменуемых проходивших экзамен, прошедших и непрошедших экзамен за определенный промежуток времени.

Алгоритм работы системы:

1. Пользователь регистрируется в базе ГИБДД для прохождения экзамена;

2. Система присваивает номер идентификации пользователю для входа с целью получения информации;
3. Система автоматически переводит пользователя на запись на нужный этап;
4. Пользователь выбирает удобную дату и время для прохождения экзамена;
5. Система формирует список экзаменуемых на основе данных о дате и времени;
6. В таблице списка предоставляется информация о ФИО, серии и номере паспорта и номер идентификации (для администратора: сотрудника гаи);
7. При нажатии на кнопку «начать тестирование» пользователю засчитывается +1 попытка;
8. После успешного прохождения этапа, в базе данных этап получает статус «Сдан» и автоматически переводит пользователя на запись на следующий этап. В ином случае в графе этапа в базе данных у пользователя будет ставиться количество попыток, а на экране пользователь получит сообщение о возможной дате пересдачи этапа.

Объектом автоматизации являются процессы по планированию экзамена в ГИБДД УМВД, а также контроль эффективности выполнения указанных процессов.

Процессы планирования экзамена включают в себя:

1. Планирование расписаний сдачи каждого из этапов экзамена;
2. Запись пользователя на удобное время;
3. Общая база экзаменуемых;
4. База экзаменуемых по каждому проведенному этапу;
5. Доступ пользователю к информации относительно себя;
6. Ведение архива с ограничением срока давности.

Для успешной организации деятельности существует необходимость:

1. Иметь постоянно обновляемую информацию об экзаменуемых и актуальных вопросах проведения экзамена;
2. Обеспечить оперативную, максимально удобную для экзаменуемого и сотрудника ГИБДД работу АС;
3. Получение в любой момент информацию о текущем состоянии экзамена (на каком этапе находится экзаменуемый, о сроках пересдачи);
4. Автоматизация этих задач позволит повысить уровень централизации планирования экзамена, группировать зону обслуживания, повысить качество приема и сдачи экзамена, в конечном итоге повысить эффективность работы.

1.3.3 Требования к системе

АСПЭ предназначена для автоматизации обмена информацией между объектами автоматизации и процессами обработки заявок внутри объектов автоматизации. Автоматизация подлежит операции подготовки, регистрации, отслеживание статуса этапа экзамена на рабочих местах пользователей системы.

Требование к режимам функционирования:

АС должна обеспечивать работу в двух режимах:

- сетевой режим взаимодействия;
- автономный режим взаимодействия.

Требование надёжности:

1. Время восстановления работоспособности автоматизированной системы при различных сбоях и отказах не должно превышать одного рабочего дня, за исключением случаев неисправности серверного оборудования;
2. В автоматизированной системе должна быть обеспечена корректная обработка сбоев электронно-механических устройств (например, принтеров,

сканеров) при выполнении функций, связанных с формированием твердых копий документов;

3. В автоматизированной системе должна быть обеспечена возможность «горячей» замены сбойного или вышедшего из строя активного накопителя на жестком магнитном диске (серверного оборудования АС) без остановки функционирования и потерь информации;

4. В АС должна быть обеспечена возможность восстановить данные с внешнего накопителя после восстановления активного накопителя. Конкретный состав требований по восстановлению данных дополняется соответствующими требованиями на подсистемы;

5. В АС должно осуществляться разграничение прав доступа к системе;

6. В АС должен вестись дневник событий системы;

7. Импульсные помехи, прекращение электропитания или сбои не должны приводить к выходу из строя технических средств АС, которые находятся в специально оборудованном помещении и подключенных к системе бесперебойного электроснабжения, в том числе автономного. Конкретный состав требований по защите оборудования от импульсных помех, прекращения электропитания и сбоев дополняется соответствующими требованиями на подсистемы;

8. В автоматизированной системе всех уровней необходимо, чтобы были реализованы функции корректной автоматической остановки работы технических средств, подключенных к системе бесперебойного электроснабжения, в том числе автономного, при длительном отсутствии электропитания.

Требование к эксплуатации, техническому обслуживанию и хранению:

1. Автоматизированная система должна обеспечивать круглосуточный, и что очень важно, непрерывный режим эксплуатации с учетом времени на техническое обслуживание;

2. В помещениях, которые предназначены для эксплуатации АС, необходимо отсутствие агрессивной среды; массовая концентрация пыли в

воздухе должна быть не более 0,75 мг/м³, электрическая составляющая электромагнитного поля помех не должна превышать 0,3 в/м в диапазоне частот от 0,15 до 300,00 МГц;

3. Требования по обеспечению пожарной безопасности и электробезопасности (заземление) в помещениях должны быть выполнены в соответствии с ГОСТ 12.1.004-91 «ССБТ. Пожарная безопасность. Общие требования»;

4. Напряжение питания сети должно составлять 220В;

5. Нормальными климатическими условиями эксплуатации системы являются:

1) относительная влажность окружающего воздуха 60 % при учете, что атмосфера воздуха будет составлять 20 ± 5 °С .

2) температура окружающего воздуха 20 ± 5 °С (система должна сохранять работоспособность при температуре воздуха от 10 до 35 °С);

2 РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ

Во второй главе мы будем проектировать структуру базы данных, структуру WEB-приложения и программную реализацию автоматизированной системы.

2.1 Проектирование структуры базы данных

На начальном этапе проектирования автоматизированной системы необходимо проектирование структуры базы данных для хранения и обработки информации.

Для разработки структуры базы данных мы выбрали инструменты ERwin и IVExpert.

IVExpert - это оболочка, которая предназначена для разработки баз данных с использованием InterBase и Firebird, а также для выбора и изменения информации, хранившейся в базе. IVExpert обладает большим количеством компонентов, которые облегчают работу: визуальный редактор для всех объектов базы данных, SQL-редактор и исполнитель скриптов, дизайнер базы, отладчик хранимых процедур и триггеров.

ERwin - средство разработки структуры базы данных (БД). ERwin сочетает в себе графический интерфейс Windows, инструменты для построения ER-диаграмм, редакторы для создания логического и физического описания модели данных и прозрачную поддержку ведущих реляционных СУБД и настольных баз данных. С помощью ERwin можно создавать проектирование баз данных.

ERwin создает визуальную модель данных для задачи, которую необходимо решить. В ERwin диаграмма строится из трех основных блоков - сущностей, атрибутов и связей.

Сущность - множество индивидуальных экземпляров, которые не повторяются. Каждая сущность обладает атрибутами.

Атрибут - свойство объекта, которое характеризует его экземпляр.

Процесс построения информационной модели данных:

1. определение сущностей;
2. определение зависимостей между сущностями;
3. задание первичных и альтернативных ключей;
4. определение атрибутов сущностей;
5. приведение модели к требуемому уровню нормальной формы;
6. переход к физическому описанию модели: назначение соответствий имя сущности - имя таблицы, атрибут сущности - атрибут таблицы; задание триггеров, процедур и ограничений;
7. генерация базы данных.

В ERwin существуют два уровня представления и моделирования, которые являются общепринятым видом изображения модели данных – логический (рис. 2.1) и физический (рис. 2.2).

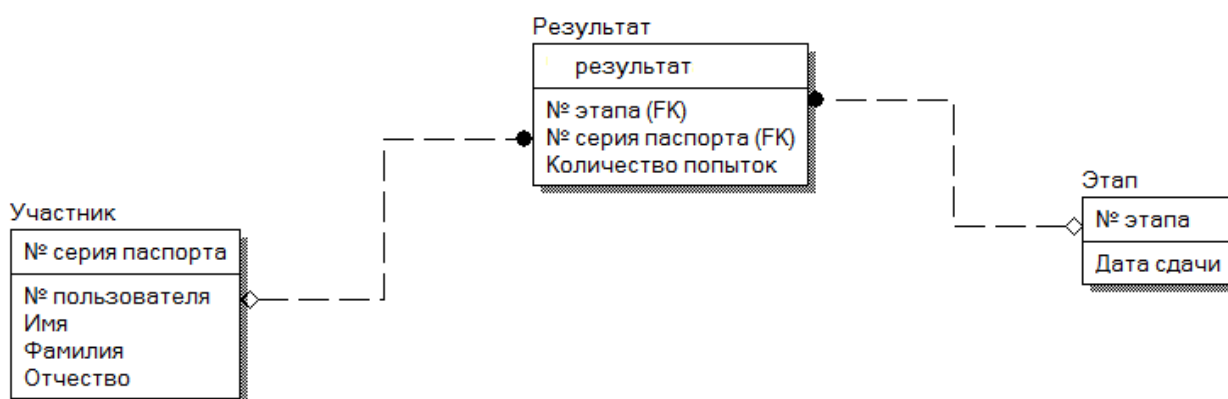


Рис. 2.1. Логическая модель базы данных

Физическая структура базы данных является отображением предметной области. Проектирование физической структуры базы данных включает в себя определение всех информационных единиц базы данных и связей, определение имен, типов и других важных характеристик.

Атрибуты, хранившие в себе информацию о ФИО, будут иметь тип VARCHAR(20) - значит ввод символов ограничен двадцатью знаками. Для серии и номера паспорта, а также номера пользователя мы задали тип INTEGER. Атрибуты «№ этапа» и «Количество попыток» будут иметь тип SMALLINT.

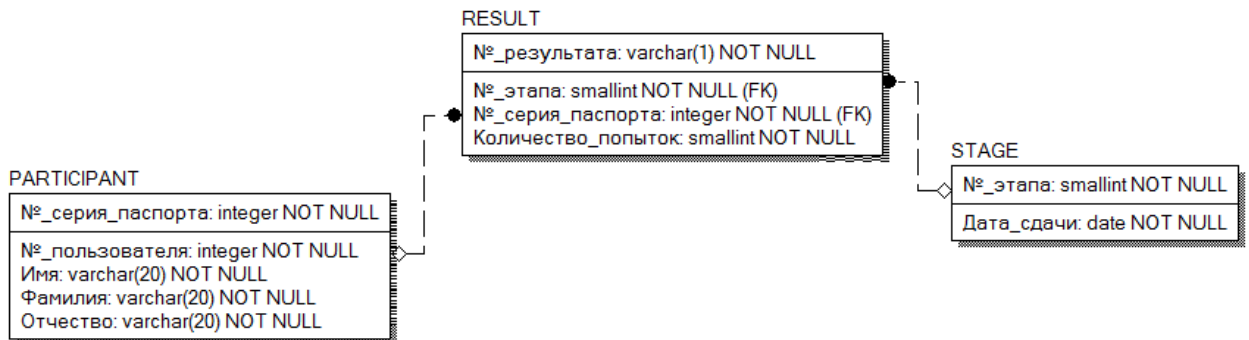


Рис. 2.2. Физическая модель базы данных

Выбором системы управления баз данных (СУБД) является СУБД Firebird Firebird - полнофункциональная и мощная СУБД, показывающая хорошую производительность и практически не нуждается в обслуживании.

Основными положительными характеристиками Firebird являются:

- 1) удовлетворяет требованию целостности и надежности транзакций;
- 2) позволяет серверу обрабатывать разные версии записи так, чтобы каждая из транзакций видела только свою запись;
- 3) позволяет создать сложную хранимую процедуру, триггер, генератор;
- 4) дает возможность множество одновременных транзакций;
- 5) обеспечивает целостность между таблицами;
- 6) позволяет назначить несколько триггеров для одной таблицы, а также внести начальные значения.

Firebird включает в себя набор консольных программ, позволяющих создавать базы данных, исследовать их характеристики, выполнить операторы SQL и скрипты, производить резервное копирование данных, их восстановление из резервной копии.

Сущность «PARTICIPANT» имеет 5 экземпляров: имя, фамилия, отчество, номер и серия паспорта, номер пользователя (листинг 2.1). Эта сущность предназначена для хранения информации о пользователе.

Листинг 2.1. Создание сущности «PARTICIPANT»

```
CREATE TABLE PARTICIPANT (
    "Имя"          VARCHAR(20) NOT NULL,
    "Фамилия"      VARCHAR(20) NOT NULL,
    "Отчество"     VARCHAR(20) NOT NULL,
    "Номер и серия паспорта" BIGINT NOT NULL,
    "Номер пользователя"  INTEGER NOT NULL);
```

Конец листинга.

Для изменения ссылочной целостности в таблицах мы использовали триггер (листинг 2.2) – действия, которые должны быть выполнены для зависимой таблицы при изменении соответствующего ключа в родительской таблице.

Листинг 2.2. Создание триггера

```
CREATE OR ALTER TRIGGER PARTICIPANT_BI FOR PARTICIPANT
ACTIVE BEFORE INSERT POSITION 0
as begin
    if (new."Номер пользователя" is null) then
        new."Номер пользователя" = gen_id(gen_numident_id,1);
    end
```

Конец листинга.

Сущность «STAGE» имеет 3 экземпляра: этап, номер и серия паспорта, дата сдачи (листинг 2.3). Эта сущность предназначена для хранения информации об этапах экзамена.

Листинг 2.3. Создание сущности «STAGE»

```
CREATE TABLE STAGE (
```

```
"Дата сдачи"    DATE NOT NULL,  
"Этап"          SMALLINT NOT NULL,  
"номер паспорта" BIGINT NOT NULL);
```

Конец листинга.

Для создания внешних ключей была использована функция «ON DELETE/ UPDATE CASCADE» (листинг 2.4), для обеспечения целостности БД для которой при удалении записи в родительской сущности автоматически удаляет запись в дочерней.

Листинг 2.4. Функция «ON DELETE/ UPDATE CASCADE»

```
ALTER TABLE STAGE ADD CONSTRAINT FK_STAGE_1 FOREIGN  
KEY ("номер паспорта") REFERENCES PARTICIPANT ("Номер и серия  
паспорта") ON DELETE CASCADE ON UPDATE CASCADE;
```

Конец листинга.

2.2 Проектирование структуры WEB – приложения

В 21 веке, веке технологий, обрел огромную популярность класс приложений - Web-приложения. Обеспечивая доступ через интернет к автоматизированным системам и базам данных, Web-приложения являются одним из самых эффективных инструментов ведения современного обмена и обработки информации.

Принцип и архитектура работы Web-приложения отличаются от стандартных программ, работающих на ПК или серверах локальной сети. Web-приложение всегда должно работать в многопоточных и многопользовательских режимах.

Как правило, Web-приложение и различные программы, ориентированные на интернет, предполагают взаимодействие клиентской программы, таких как браузер, и сервера.

Разработку Web-приложения следует начать с разработки проекта. На этом этапе необходимо выяснить, что конкретно требуется сделать и каким образом.

Веб-приложения, разработанное нами состоит из трёх основных компонентов (рис.2.3):

1. Клиентская часть Web - приложения - это графический интерфейс, то, что пользователь увидит на Web - странице. Графический интерфейс отображается в браузере. Пользователь взаимодействует с Web-приложением непосредственно через браузер, кликая по ссылкам и кнопкам. Дизайн определяет внешний вид Web- приложения, в котором оно предстанет перед пользователями.

2. Серверная часть Web - приложения - это программа или скрипт на сервере, обрабатывающая запросы. Серверную часть Web - приложения мы программировали на PHP. При каждом переходе пользователя по ссылке браузер отправляет запрос к серверу. Сервер обрабатывает этот запрос, вызывая некоторый PHP-скрипт, который формирует Web - страницу, описанную языком HTML, и отправляет клиенту по сети. Браузер тут же отображает полученный результат в виде всплывающего окна на Web – странице.

3. База данных (БД, или система управления баазами данных, СУБД) - программное обеспечение на сервере, занимающееся хранением данных и их выдачей в нужный момент. База данных располагается на сервере. Серверная часть веб-приложения (то есть, PHP скрипт) обращается к базе данных, извлекая данные, которые необходимы для формирования страницы, запрошенной пользователем.

Алгоритм взаимодействия компонентов

- 1) Браузер через Интернет отправляет HTTP-запросы веб-серверу;
- 2) Веб-сервер вызывает PHP-скрипт, написанный разработчиком веб-приложения;
- 3) PHP-скрипт обращается к базе данных, если это нужно;

4) В результате PHP-скрипт возвращает клиенту веб-страницу, которую и отображает браузер.

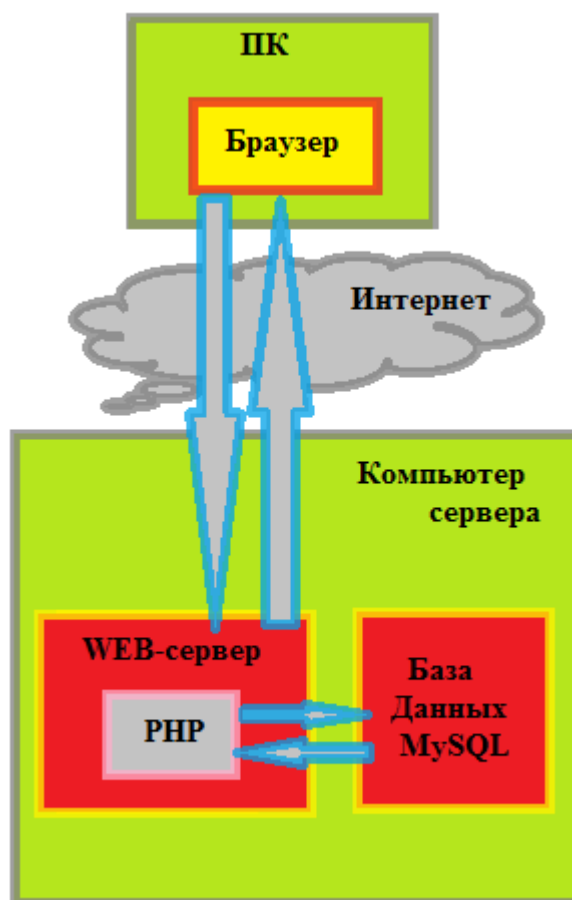


Рис. 2.3. Взаимодействие компонентов приложения

Для разработки Web-серверов, являющихся Web-приложениями, будем использовать язык разметки гипертекста HTML. То, что видят пользователи Web-сервера в окне браузера, представляет собой документ, составленный на языке HTML и содержащий внутри себя объекты различных типов (изображения, анимацию, формы для ввода информации).

Вэб приложение нашей автоматизированной системы должно состоять из одной вэб страницы, где содержится вся необходимая информация о порядке сдачи экзамена, передаче, категориях и регистрации пользователей. Переход по странице пользователь должен осуществить прокруткой колеса мыши или кликая на нужный раздел в главном меню. Так же необходима контактная информация в случае возникновения вопросов у пользователя.

Для удобства восприятия форм для заполнения пользователем, в окне формы необходимо «подсветить» пример или действие, необходимое к выполнению.

Вся выходная информация должна осуществляться внутри одной существующей главной страницы с помощью всплывающих окон.

Доступ к автоматизированная информационной системе необходимо обеспечить с любых ПК и мобильных устройств, оборудованных возможностью выхода в интернет.

2.3 Программная реализация

Для программирования серверной части web-приложения мы используем язык PHP. Для разработки web-приложений мы также используем язык программирования Java.

Если web-серверу необходимо кроме вывода информации пользователям, еще и получать информацию на вход, будь то данные или файл, нужно создать «активное» web-приложение. Кроме статической страницы HTML, подобное приложение может содержать компоненты, динамически создающие страницы HTML.

HTML - основной язык, которым описывается графический интерфейс веб-приложения. Данный язык описывает структуру веб-страницы. Художественное оформление веб страниц, как правило, описывается таблицами стилей - CSS.

HTML – стандартный язык разметки документа в сети интернет. Большинство Web – страниц созданы при помощи языка разметки HTML (листинг 2.5). Данный язык HTML выполняется браузерами, в следствии чего на экране ПК или переносного устройства пользователь может видеть отформатированный текст.

Листинг 2.5. Пример кода на языке HTML

```
<div class="modal-overlay" id="modal-overlay">
```

```

<div class="modal" id="modal">
  <button class="close-button" id="close-button">Отменить</button>
  <div class="modal-guts">
    <h1>Выбор Даты</h1>
    <style>
.calendar3 {
width: 100%;
font: monospace;
line-height: 1.2em;
font-size: 15px;
text-align: center;
}; </style></div></div></div>

```

Конец листинга.

Также, существует язык XHTML - наиболее строгий вариант HTML, соответствующий всем ограничениям XML и, фактически, XHTML можно воспринять как приложение языка XML к области разметок гипертекстов.

В интернете HTML-страницы, как правило, передаются в виде простого текста или с использованием шифрования браузерам от сервера по протоколу HTTP/ HTTPS

PHP – распространенный язык программирования общего назначения с открытым исходным текстом. Язык PHP создан специально для ведения web-разработок. PHP скрипт можно внедрить напрямую в HTML.

Вместо стандартного вывода HTML-кода командами языка, текст PHP содержит HTML с вкраплениями кода. Код PHP отделяется специальными начальными «<? php» и конечными «?>» (листинг 2.6) тэгами, которые позволяют подключиться PHP-режим и выходить из него.

Листинг 2.6. Пример кода на языке PHP

```

<?php
$imy = $_POST['imy'];

```



```
$familia = $_POST['familia'];  
$otchestvo = $_POST['otchestvo'];  
$pasport = $_POST['pasport'];  
?>
```

Конец листинга.

Различия PHP и JavaScript состоит в том, что PHP-скрипты реализуются на сервере и генерируют HTML, который посылается пользователю. Если скрипт размещен на сервере, пользователь получит только результат его выполнения и не сможет понять какой именно код вывел информацию. Так же, можно настроить сервер так, чтобы обычные HTML-файты обрабатывались процессором PHP таким образом, что пользователи не смогут узнать, получают они обычные HTML-файлы или результаты выполнения скриптов.

PHP очень простой в освоении. PHP имеет длинный список возможностей.

Предназначением PHP является работа в среде web-серверов, но тем не менее, область его применения не ограничивается только этим.

Java — более типизированный объектно-ориентированный язык программирования. Приложение на Java обычно транслируется в специализированный байт-код, поэтому оно может работать на любом компьютере, с помощью виртуальных Java-машин.

Достоинством одного из способов выполнения программы является полная независимость байт-кода от операционной системы и оборудования. Это позволяет выполнять Java-приложения на любом устройстве, для которого имеется соответствующая виртуальная машина. Еще одной важной особенностью технологии Java является гибкая система безопасности, которая полностью контролирует исполнение программы в виртуальной машине. Любые операции, превышающие установленные рамки программы, вызовут немедленную остановку.

Недостатками концепции виртуальных машин является понижение производительности. Несколько улучшений позволили увеличить скорость выполнения программы на Java:

1) применение технологий трансляции байт-кода в машинный код непосредственно во время работы программы (JIT-технология) с возможностью сохранения версий класса в машинном коде,

2) широкое использование платформенно-ориентированного кода (native-код) в стандартных библиотеках,

3) аппаратные средства, обеспечивающие ускоренную обработку байт-кода (например, технология Jazelle, поддерживаемая некоторыми процессорами фирмы ARM).

Для 7 различных задач время выполнения на Java составит в среднем в 1,5-2 раза больше, чем для C/C++, в некоторых случаях Java быстрее, а в отдельных случаях в 7 раз медленнее. С иной стороны, для большинства из этих задач потребление памяти Java-машиной было в среднем в 10—30 раз больше, чем программой на C/C++. Также замечено, анализ, проведённый компанией Google, согласно которому отмечается существенно более низкая производительность и большее потребление памяти в тестовых примерах на Java в сравнении с аналогичными программами на C++.

JavaScript - язык программирования, с поддерживает объектно-ориентированных, императивных и функциональных стилей (листинг 2.7.).

JavaScript обычно используют в программе для доступа к объектам приложений. В основном JavaScript применяется в браузерах для придания интерактивности веб-страницам как язык сценариев.

Листинг 2.7. Пример кода на языке JavaScript

```
var modal = document.querySelector("#modal");
var modalOverlay = document.querySelector("#modal-overlay");
var closeButton = document.querySelector("#close-button");
var openButton = document.querySelector("#open-button");
closeButton.addEventListener("click", function() {
```

```
modal.classList.toggle("closed");
modalOverlay.classList.toggle("closed");
});
openButton.addEventListener("click", function() {
    modal.classList.toggle("closed");
    modalOverlay.classList.toggle("closed");
});
```

Конец листинга.

Основными архитектурными чертами являются: динамические типизации, слабые типизации, автоматическое управление памятью.

Достаточно много языков программирования оказывали влияние на JavaScript. Во время разработки была поставлена цель: создать язык похожий на Java, но более лёгкий в использовании непрограммистами. Отличие языка JavaScript от других языков программирования, используемых при web-разработке в том, что этим языком не владеют компании и организации. Название «JavaScript» зарегистрировано товарным знаком компании Oracle Corporation.

JavaScript является объектно-ориентированным языком, но используемые языком прототипирования обуславливают различия в работе с объектами в сравнении с традиционными класс-ориентированными языками. Более того, JavaScript имеет свойства, присущие функциональным языкам — функции как объекты первого класса, объекты как списки, карринг, анонимные функции, замыкания — что придаёт языку дополнительную гибкость.

Несмотря на похожий синтаксис с языком программирования Си, JavaScript в сравнении с языком Си имеет кардинальные различия:

- 1) объект с возможностью интроспекции;
- 2) функция как объект 1 класса;
- 3) автоматическое приведение типов;

- 4) автоматическая сборка мусора;
- 5) анонимные функции.

В языке JavaScript отсутствуют некоторые полезные вещи:

- 1) модульная система: JavaScript не предоставляет возможностей управлять зависимостями и изоляцией области видимости;
- 2) стандартная библиотека: отсутствует интерфейс программирования приложений в работе с файловой системой, управлением потоками ввода-вывода, базовых типов для бинарных данных;
- 3) стандартные интерфейсы к web-серверам и базам данных;
- 4) система управления пакетами, отслеживающая зависимость и автоматически устанавливает ее.

CSS — формальный язык, описывающий внешний вид документа, написанного с использованием языка разметки.

В большинстве случаев используется как средство описания и оформления внешнего вида web-страниц, написанных с помощью языка разметки HTML или языка XHTML. Так же CSS применяется к любым XML-документам.

CSS используется создателями web-страниц для задания шрифтов, цветов, расположения отдельных блоков и других аспектов представления внешнего вида этой web-страницы. Основная цель разработки CSS - разделить описание логической структуры web-страницы от описания внешнего вида web-страницы. Подобное разделение может значительно повысить доступность документа, предоставить возможность управления его представлением и большую гибкость, а также уменьшить повторяемость в структурном содержимом и сложность. Вдобавок, CSS позволяет представлять один и тот же документ в самых разных стилях или методах вывода, например, таких как печатное представление, экранное представление.

Листинг 2.8. Пример кода на CSS

```
.open-button {
```

```
border: 0;
position: absolute;
top: 80%;
left: 90%;
transform: translate(-50%, -50%);
background: lightgreen;
color: white;
padding: 10px 20px;
border-radius: 10px;
font-size: 21px;
}
```

Конец листинга.

Для «оживления» графического интерфейса можно использовать дополнительные технологии: скрипт JavaScript, а также встроенные в web-страницу компоненты. Все элементы web-страниц могут взаимодействовать между собой: программы, написанные на JavaScript и выполняющиеся на web-страницах, могут управлять встроенными в страницы компонентами, тем самым реализуя пользовательский интерфейс с большими возможностями.

Страницы, которые созданы лишь с применением языка HTML - статичны. У подобных страниц нет возможности взаимодействия с пользователями, получения информации, так же нет возможности ее обрабатывать. Статические страницы HTML обычно содержат только текст, изображения и ссылки, предназначенные для перехода на другие страницы,

Для «оживления» нашего web-приложения, кроме анимационных графических изображений, которые состоят из нескольких сменяющихся кадров, требуется прибегать к программированию на языках сценариев - JavaScript или Visual Basic Scripting Edition.

Следующий этап освоения Web-технологий необходимо начинать с изучения наиболее распространенного языка сценариев JavaScript, а также динамического языка разметки страниц DHTML (Dynamic HTML). Далее следует ознакомиться с апплетами Java, которые придают динамичность страницам HTML и которые пригодны для решения ряда других задач.

Клиентские сценарии размещаются внутри Web-страниц и запускаются (как интерпретируемые программы под управлением браузера) на компьютере посетителя Web-сайта. Также существуют и серверные сценарии; о них мы расскажем ниже.

Чтобы приступить к созданию сложных Web-приложений недостаточно освоить языки JavaScript, DHTML и апплеты Java. Тем не менее, этап изучения этих языков достаточно важен для дальнейшего изучения.

Как правило, разные браузеры по-разному интерпретируют клиентские сценарии, что приводит к появлению непростой проблемы совместимости. Для серверных сценариев, работающих под управлением Web-сервера, таких проблем не существует. Третий этап программирования предполагает изучение технологий, которые позволяют получать данные от посетителей Web-приложения и обрабатывать их на Web-сервере, при необходимости обращаясь к СУБД. На этом этапе следует изучить активные серверные страницы ASP (Active Server Pages), которые применяются в Web-сервере Microsoft Internet Information Server (IIS) для создания сложных интерактивных Web-приложений.

Страницы ASP содержат серверные сценарии на языках JavaScript или VB Script. Для того, чтобы успешно создавать страницы ASP, нам необходимо освоить один из этих языков. Применение VB Script достаточно хорошо освещено в литературе и прежде всего в документации, поставляемой Microsoft. Кроме того, вам необходимо изучить средства создания страниц ASP, которые предоставляет программисту сервер IIS.

Почти любое автоматизированное информационные Web-приложение хранит информацию в базах данных. Мы расскажем об использовании серверов

СУБД Microsoft SQL Server 7.0 и SQL Server 2000, применяемых на платформе Microsoft Windows NT 4.0 и Windows 2000 Server. На сайтах, созданных с применением широко распространенной операционной системы Windows популярны подобные бесплатные или практически – бесплатные СУБД как MySQL.

Для доступа к данным, хранящимся в СУБД, используются специальные языки программирования. Например, в СУБД Microsoft SQL Server применяется язык Transact-SQL.

Технология ADO предоставляет в распоряжение программиста простой и удобный набор объектов и интерфейсов, позволяющих эффективно выполнять все операции с базами данных из сценариев ASP.

3 ИСПЫТАНИЕ

В третьей главе мы будем рассматривать методику испытания нашей автоматизированной системы и результаты ее тестирования.

3.1 Программа и методика испытаний

Объектом испытаний является автоматизированная система планирования экзамена в ГИБДД. Методика испытаний основана на требованиях и функциях, выдвинутых к автоматизированной информационной системе в ходе разработки.

Первым делом необходимо осуществить вход на страницу автоматизированной системы.

После входа на страницу необходимо ознакомиться с главным меню и с предоставленной информацией.

Далее следует осуществить движение по странице как с помощью колеса мыши, так и переходом по ссылкам главного меню.

Следующим этапом тестирования автоматизированной системы будет функция регистрации пользователей. Напомним, что в эту функцию входит информация:

1. ФИО экзаменуемого
2. Серия и номер паспорта экзаменуемого
3. Выбор удобной для экзаменуемого даты и времени прохождения экзамена

Для ввода необходимой информации пользователю нужно заполнить предоставленную обязательную форму, после чего нажать кнопку «Зарегистрироваться»

После нажатия кнопки регистрации, во всплывающем окне, экзаменуемому предоставляется информация о регистрации и присвоении

номера. Для закрытия окна и продолжения работы на странице, предлагается кнопка «Закрыть».

Далее пользователю необходимо записаться. Для записи необходимо нажать кнопку «Записаться». Для записи на этап пользователю необходимо ввести серию и номер паспорта в окне web-страницы. Далее появится информация о дате. Кликом кнопки мыши по удобной дате пользователь имеет возможность выбора даты. Для прохождения к следующему этапу выбора времени пользователю необходимо нажать кнопку «Далее». Во всплывающем окне появится информация о возможном времени для записи. Так же как и в предыдущем этапе, кликом мыши экзаменуемый выбирает необходимое время и нажимает кнопку «Далее».

Для получения информации пользователю необходимо нажать кнопку «Получить информацию» на web - странице. Далее пользователю необходимо авторизоваться путем ввода серии и номера паспорта.

3.2 Результаты испытаний

Результат входа на главную страницу автоматизированной системы планирования экзамена в ГИБДД УМВД со стационарного ПК представлен на рис. 3.1.

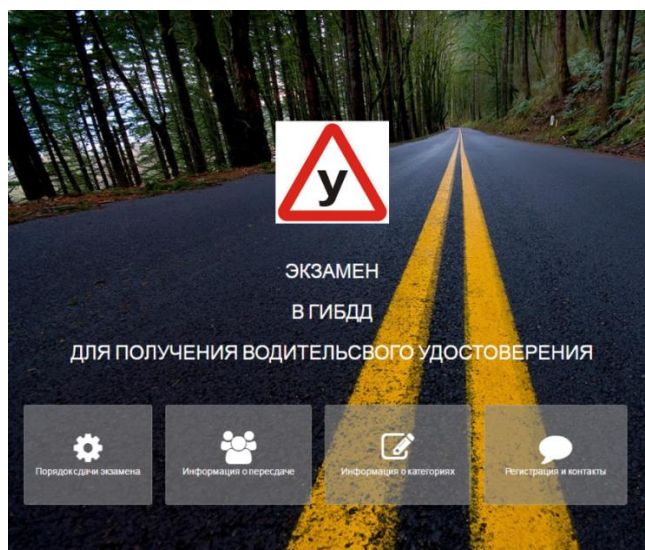


Рис. 3.1. Страница автоматизированной системы

Результат входа на главную страницу автоматизированной системы планирования экзамена в ГИБДД УМВД с мобильного телефона представлен на рис. 3.2.

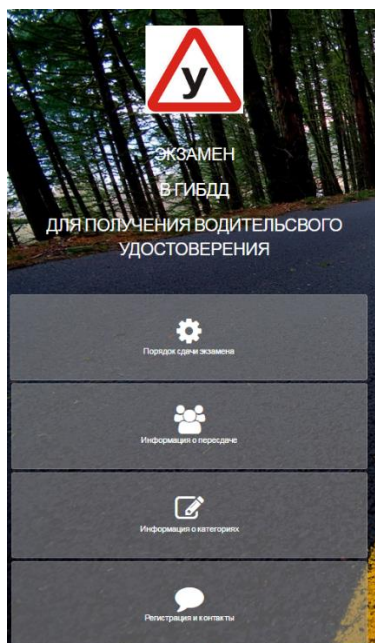
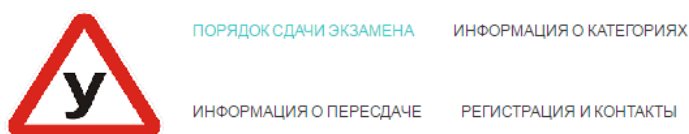


Рис. 3.2. Страница АС с мобильного телефона

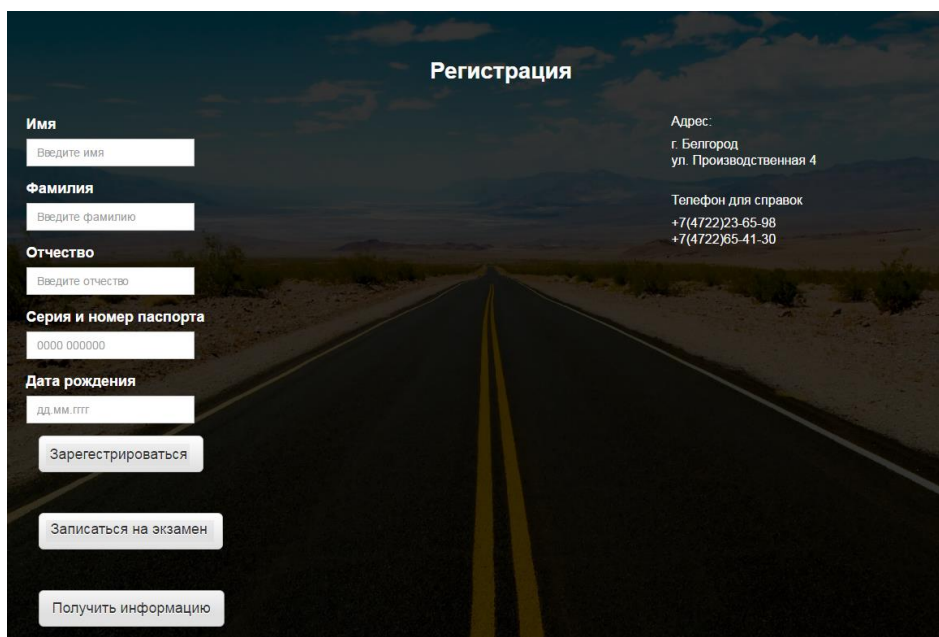
Результат ознакомления с информацией на странице автоматизированной системы планирования экзамена в ГИБДД УМВД представлен на рис. 3.3.



Допуск к сдаче экзаменов на получение водительского удостоверения в ГИБДД производится, если вы прошли обучение в организации, имеющей лицензию на такое обучение (на категории «А» и «В» допускалась самостоятельная подготовка до 5 ноября 2013 года). С 2014 года с 16 лет можно открыть категории «А1» и «М»; категории «А», «В» («В1») и «С» («С1») с 18 лет; категория «D» («D1») доступна с 21 года (при условии открытой категории «С» («С1»)); для «прицепных» категорий «ВЕ», «СЕ» («С1Е») или «DE» («D1E») необходимо иметь стаж вождения тягача соответствующей категории не менее 1 года. Экзамен состоит из трех этапов: теория правил дорожного движения (ПДД), автодром (площадка) и город. Тестирование теоретической части производится на компьютерах. На решение теоретической части отводится 20 минут. Тест состоит из 20 вопросов одного билета. Допускается сделать 2 ошибки. В отличие, от различных тренировочных программ, при сдаче экзамена в ГИБДД, у экзаменуемого нет возможности узнать верность ответа на каждый вопрос. Все свои ошибки и их количество экзаменуемый узнает только после ответа на все 20 вопросов билета. При положительном результате сдачи теоретической части далее следует второй этап – площадка. Сотрудник ГИБДД назначает экзаменуемому три упражнения на его выбор, из тех четырех, что отработывались во время обучения, это «Симметричная змейка», «Остановка и трогание на подъеме», «Параллельная парковка задним ходом» или «Разворот». Инспектор ГИБДД наблюдает за сдачей экзамена со стороны, т.е. во время сдачи экзаменуемый находится в машине один. В случае если и с этой частью экзаменуемый справился – отправляется на заключительный этап – город. Вождение по городу – третий, заключительный этап экзамена. К нему допускаются только те кандидаты, которые успешно сдали теорию и площадку. Несколько кандидатов садятся в автомобиль: один за руль, остальные – на пассажирские сиденья. Место рядом с водителем занимает экзаменатор. Он сообщает экзаменуемым, по какому маршруту нужно следовать. Список маршрутов составляется заранее, информация вывешивается на стенде в отделении ГИБДД. Кандидат проезжает какой-то отрезок пути. За каждую ошибку инспектор ГИБДД выставляет определенное количество баллов. Баллы суммируются. Если число штрафных баллов менее 5, экзаменуемый получает оценку «Сдал». При 5 и более баллах ставится оценка «Не сдал».

Рис. 3.3. Информация о порядке сдачи экзамена

Результат отображения формы регистрации на странице автоматизированной системы планирования экзамена в ГИБДД УМВД представлен на рис. 3.4.

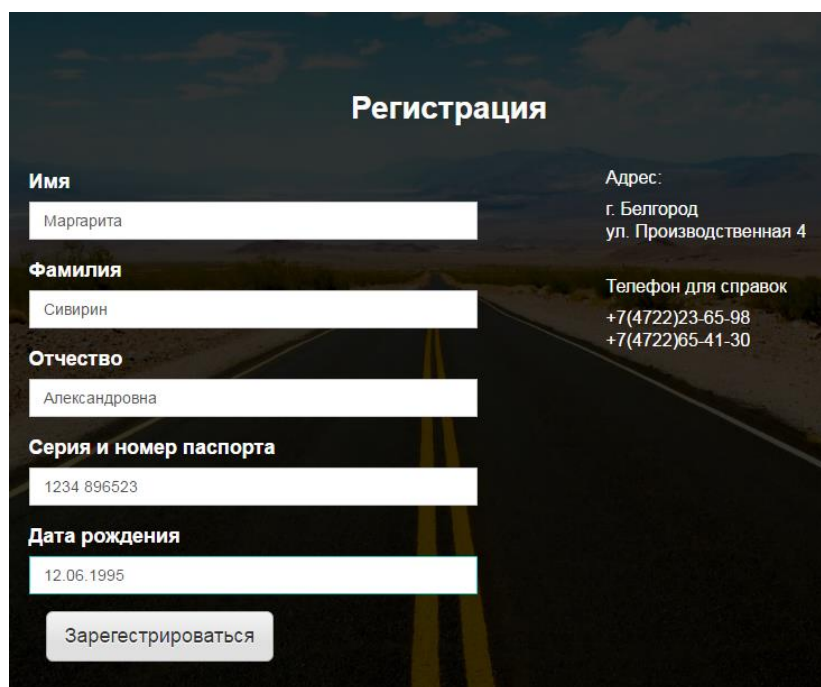


The screenshot shows a web form titled "Регистрация" (Registration) on a dark background with a road image. The form contains several input fields and buttons. The fields are labeled: "Имя" (Name), "Фамилия" (Surname), "Отчество" (Patronymic), "Серия и номер паспорта" (Passport series and number), and "Дата рождения" (Date of birth). The "Адрес" (Address) and "Телефон для справок" (Contact phone) fields are pre-filled with text. There are three buttons: "Зарегистрироваться" (Register), "Записаться на экзамен" (Sign up for exam), and "Получить информацию" (Get information).

Field Label	Value
Имя	Введите имя
Фамилия	Введите фамилию
Отчество	Введите отчество
Серия и номер паспорта	0000 000000
Дата рождения	дд.мм.гггг
Адрес	г. Белгород ул. Производственная 4
Телефон для справок	+7(4722)23-65-98 +7(4722)65-41-30

Рис. 3.4. Форма заполнения «Регистрация»

Результат ввода информации в форму регистрации на странице автоматизированной системы планирования экзамена в ГИБДД УМВД представлен на рис. 3.5.



The screenshot shows the same registration form as in Figure 3.4, but with data entered into the input fields. The "Имя" field contains "Маргарита", "Фамилия" contains "Сивирин", "Отчество" contains "Александровна", "Серия и номер паспорта" contains "1234 896523", and "Дата рождения" contains "12.06.1995". The "Адрес" and "Телефон для справок" fields remain pre-filled. The "Зарегистрироваться" button is visible at the bottom.

Field Label	Value
Имя	Маргарита
Фамилия	Сивирин
Отчество	Александровна
Серия и номер паспорта	1234 896523
Дата рождения	12.06.1995
Адрес	г. Белгород ул. Производственная 4
Телефон для справок	+7(4722)23-65-98 +7(4722)65-41-30

Рис. 3.5. Заполненная форма с данными

Результат вывода информации после регистрации на странице автоматизированной системы планирования экзамена в ГИБДД УМВД представлен на рис. 3.6.

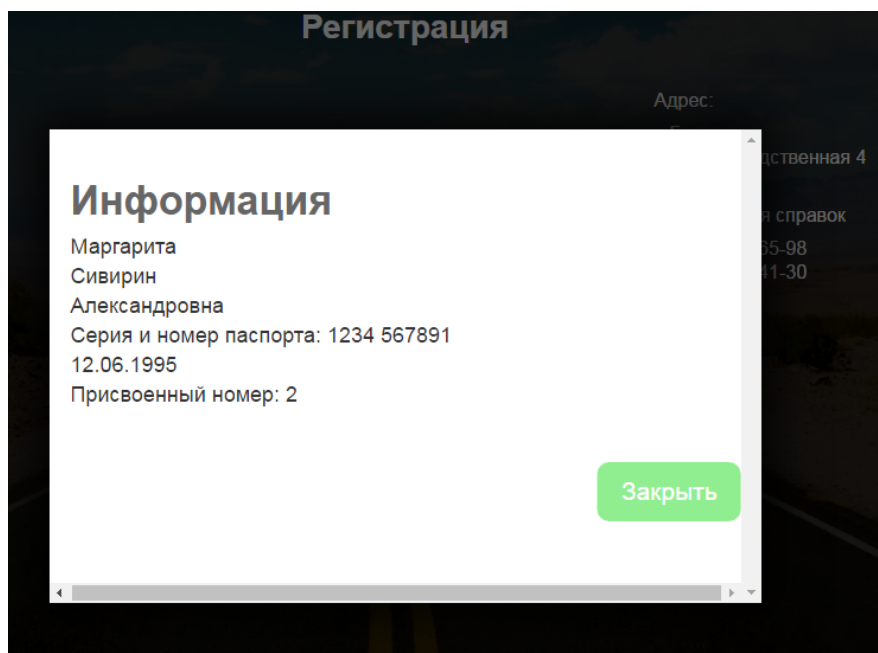


Рис. 3.6. Вывод информации после регистрации

Результат входа информации о дате на странице автоматизированной системы планирования экзамена в ГИБДД УМВД представлен на рис. 3.7.

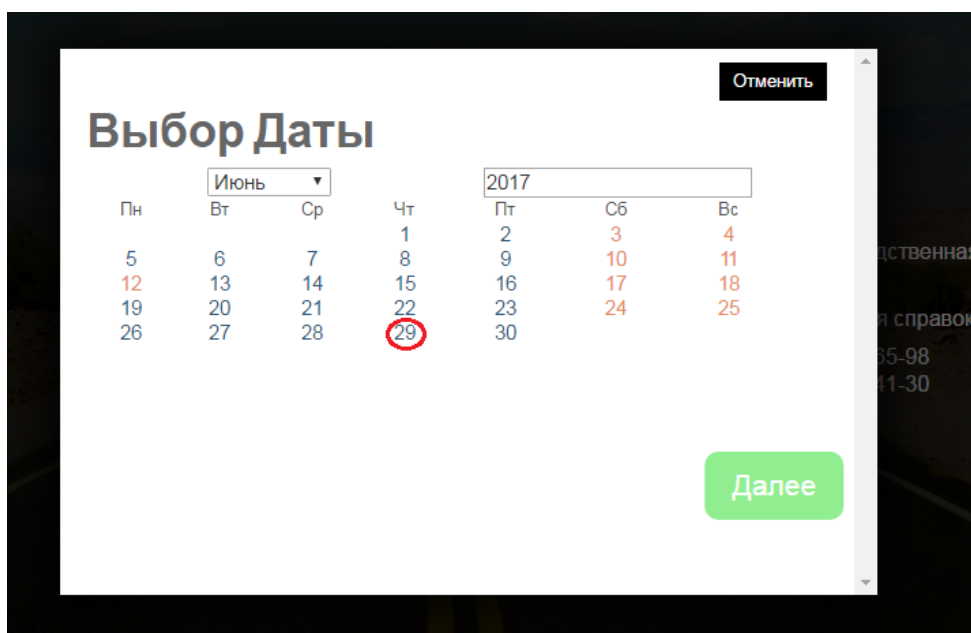


Рис. 3.7. Выбор даты

Результат входа информации о времени на странице автоматизированной системы планирования экзамена в ГИБДД УМВД представлен на рис. 3.8.

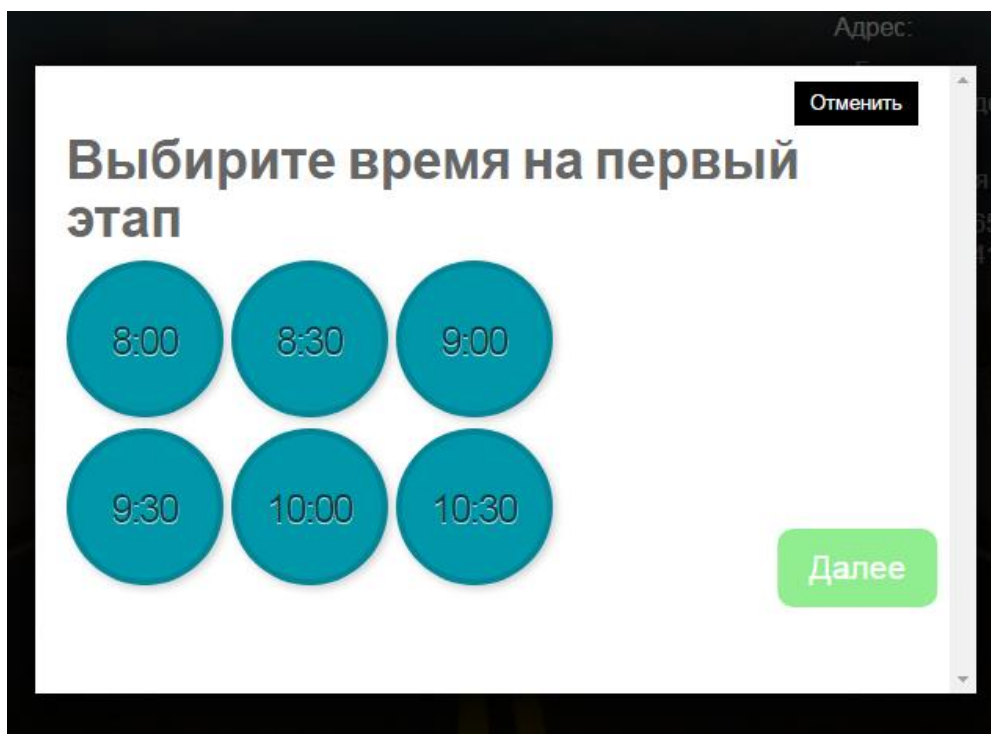


Рис. 3.8. Выбор времени

Результат формы заполнения после нажатия кнопки «Получить информацию» для получения информации о хоте экзамена пользователя на странице автоматизированной системы планирования экзамена в ГИБДД УМВД представлен на рис. 3.9.

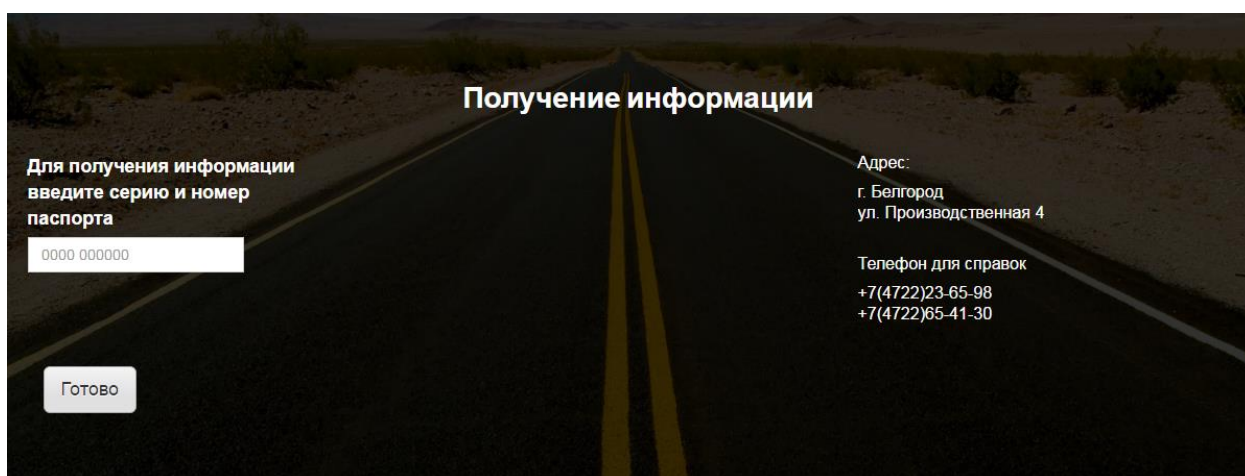


Рис. 3.9. Форма заполнения для получения информации

Результат вывода информации о ходе экзамена на странице автоматизированной системы планирования экзамена в ГИБДД УМВД представлен на рис. 3.10.

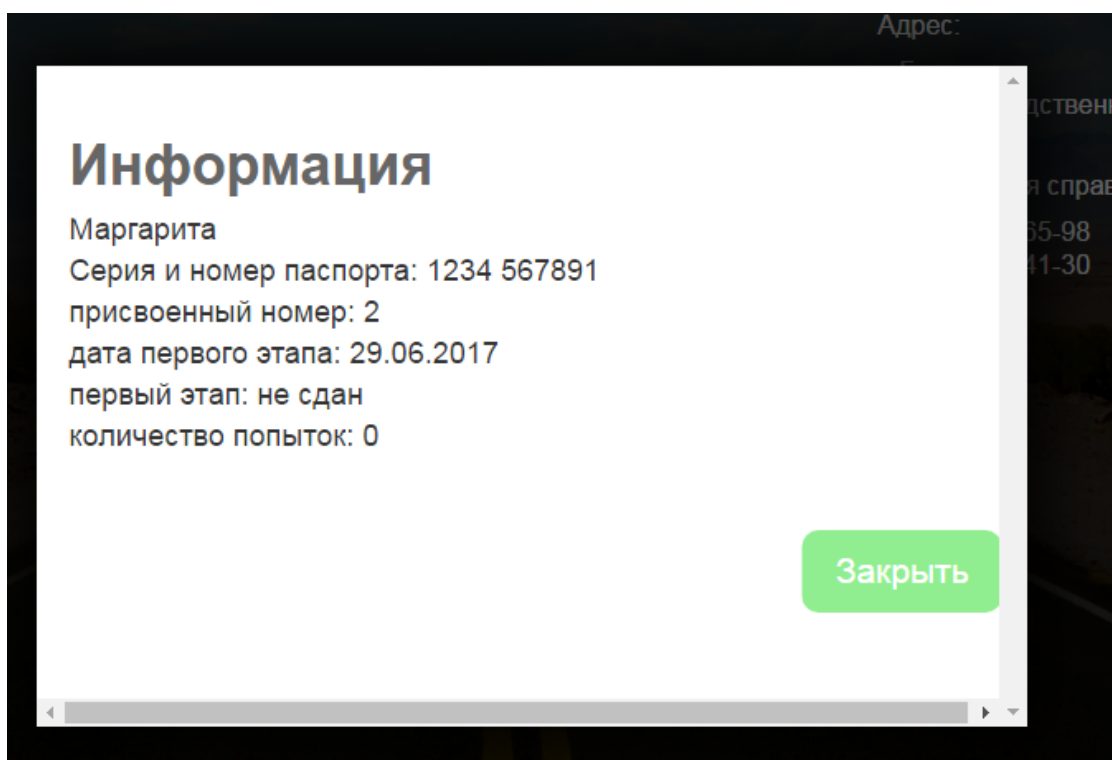


Рис. 3.10. Вывод информации

ЗАКЛЮЧЕНИЕ

В ходе разработки нашей автоматизированной системы планирования экзамена в ГИБДД УМВД мы выдвинули ряд требований к автоматизированной системе. В ходе анализа существующих систем, выполняющих подобные задачи, мы обнаружили ряд несоответствий, которые определили необходимость разработки новой автоматизированной системы, которая выполняет функцию выдачи водительского удостоверения. Автоматизированная система планирования экзамена в ГИБДД УМВД обусловлена потребностью автоматизации планирования экзамена, а именно очереди, которая улучшит процесс сдачи экзамена.

Целью выпускной квалификационной работы являлась более устойчивая автоматизация процесса сдачи экзамена и уменьшение затраченного времени.

Для достижения цели мы поставили перед собой ряд задач:

1. Обзор и анализ проведения экзамена в ГИБДД УМВД;
2. Обзор и анализ существующих систем, а именно:
 - 1) многофункционального центра предоставления государственных и муниципальных услуг;
 - 2) одно окно;
 - 3) электронной регистратуры;
 - 4) госуслуг.
3. Формулировка требований к системе;
4. Проектирование автоматизированной системы;
5. Программная реализация системы;
6. Проведение испытаний.

Успешными результатами работы автоматизированной системы планирования экзамена в ГАИ являлись:

1. Уменьшение затраченного времени на подачу документов для сдачи экзамена в ГИБДД УМВД;
2. Уменьшение затраченного времени для сдачи и приема экзамена в ГИБДД УМВД;
3. Предоставление пользователю полного доступа к информационным материалам, связанным с актуальными вопросами сдачи(пересдачи) экзамена в ГИБДД УМВД;
4. Повышение информированности пользователя по актуальным вопросам сдачи экзамена;
5. Улучшение информационного взаимодействия между пользователем и сотрудниками ГИБДД;
6. Упрощение процесса записи на все этапы для сдачи экзамена в ГИБДД.

Использование данной АС позволило сделать процесс сдачи экзамена удобнее и быстрее как для экзаменуемых, так и для сотрудников подразделений ГИБДД.

Все поставленные задачи нами были успешно выполнены.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Хомоненко А.Д., Цыганков В.М., Мальцев М.Г. Базы данных: Учебник для высших учебных заведений/Под ред. проф. А.Д. Хомоненко. – СПб.: КОРОНА принт, 2002. – 672с.
2. Емельянова Н.З. Основы построения автоматизированных систем: учеб. пособие / Н.З. Емельянова. - М.: ИНФРА-М, 2009. - 269 с.
3. Карпова Т.С. Базы данных: модели, разработка, реализация / Т.С. Карпова. - СПб.: Питер, 2007. - 304 с.
4. Клещев А.С. Методы и средства разработки пользовательского интерфейса: современное состояние / А.С. Клещев, В.В. Грибова. - СПб.: Питер, 2010. - 320 с.
5. Норенков И.П. Основы автоматизированного проектирования: Учеб. для вузов / И.П. Норенков. - М.: МГТУ им. Н.Э. Баумана, 2002. - 336 с.
6. Горев А., Эффективная работа с СУБД/ Ахаян Р., Макашарипов С.- СПб.:Питер, 1997.— 704 с.,ил.
7. Корнеев В.В.,Базы данных. Интеллектуальная обработка информации/ А.Ф. Гареев, С.В. Васютин, В.В. Райх – М.: Нолидж, 2001.- 496с.
8. Кириллов В.В.,Структурированный язык запросов (SQL). – СПб.: ИТМО, 1994. – 80 с.
9. Глушаков С.В.,Базы данных/Ломотько Д.В.. - Х.: Фолио, 2002.–504 с.
10. Шумаков П.В., Руководство разработчика баз данных/Фаронов В.В. — М.: Нолидж, 2000. — 635 с.
11. Емельянова Н.З., Основы построения автоматизированных информационных систем: учебное пособие для студ./ Партыка Т.А., Попов И.Е. - М: Форум; Инфра-М, 2011.- 416 с.

12. Кузубов В.Н., Принципы построения автоматизированных информационных систем (АИС). Техническое и рабочее проектирование АИС: учеб. пособие для вузов - М: Совр.Гум.Ун-т, 2008.-102 с.

13. Моисеенко С. SQL : задачи и решения. - СПб: Питер, 2009.-256 с.

14. В.Н.Кузубов, Основы автоматизированных информационных систем: Автоматизированные информационные системы (АИС): учеб. пособ. для вузов. - М: Совр. Гум. Ун-т, 2010.

15. Попов И.И. Информационные ресурсы и системы: реализация, моделирование, управление. –М.: ТПК АЛЬЯНС, 1996. – 408 с.

ПРИЛОЖЕНИЕ

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Untitled Document</title>
<link href='http://fonts.googleapis.com/css?family=Lato:400,700'
rel='stylesheet' type='text/css'>
<!-- Latest compiled and minified CSS -->
<link rel="stylesheet"
href="http://netdna.bootstrapcdn.com/bootstrap/3.1.1/css/bootstrap.min.css">
<!-- Optional theme -->
<link rel="stylesheet"
href="http://netdna.bootstrapcdn.com/bootstrap/3.1.1/css/bootstrap-
theme.min.css">
<!-- Latest compiled and minified CSS -->
<link rel="stylesheet" href="style.css">
<link rel="stylesheet" href="css/font-awesome.min.css">
<link rel="stylesheet/less" type="text/css" href="style.less" />
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js"></script>
<script src="js/bootstrap.min.js" type="text/javascript" ></script>
<script src="js/jquery.smint.js" type="text/javascript" ></script>
<script type="text/javascript" src="js/jquery.easing.min.js"></script>
<script type="text/javascript" src="js/jquery.easy-ticker.js"></script>
<script src="less.js" type="text/javascript"></script>
<script src="js/isotop.js"></script>

```

```

<script src="js/jquery.colorbox.js"></script>
<script type="text/javascript">
    $('.collapse').collapse();
    $(function() {
        $('a[href*=#]:not([href=#])').click(function() {
            if (location.pathname.replace(/^\/,") == this.pathname.replace(/^\/,") &&
location.hostname == this.hostname) {
                var target = $(this.hash);
                target = target.length ? target : $('[name=' + this.hash.slice(1) +']');
                if (target.length) {
                    $('html,body').animate({
                        scrollTop: target.offset().top
                    }, 1000);
                    return false;
                }
            }
        });
    });
    $(document).ready( function() {
        $('.subMenu').smint({
            'scrollSpeed' : 1000
        });
        var dd = $('.vticker').easyTicker({
            direction: 'up',
            easing: 'easeInOutBack',
            speed: 'slow',
            interval: 4000,
            height: 'auto',
            visible: 1,

```

```

mousePause: 0,
controls: {
    up: '.up',
    down: '.down',
    toggle: '.toggle',
    stopText: 'Stop !!!'
}
}).data('easyTicker');
});
$(function () { $('.tooltip-hide').tooltip('hide');});
$(document).ready(function(){
//
    var $container = $('.portfolioContainer');
    $container.isotope({
        filter: '*',
        animationOptions: {
            duration: 750,
            easing: 'linear',
            queue: false
        }
    });
    $('.portfolioFilter a').click(function(){
        $('.portfolioFilter .current').removeClass('current');
        $(this).addClass('current');

        var selector = $(this).attr('data-filter');
        $container.isotope({
            filter: selector,
            animationOptions: {
                duration: 750,

```

```

        easing: 'linear',
        queue: false
    }
});
return false;

});
$(".group2").colorbox({rel:'group2', transition:"fade"});
});
$('.carousel').carousel();
</script>
</head>
<body>
<div class="wrap ">
    <div class="section sTop">
        <div class="top-container">
            <div class="container">
                <div class="large-logo">
                    <a href="#"></a>
                </div>
                <div class="vticker">
                    <ul>
                        <li>ЭКЗАМЕН</li>
                        <li>В ГИБДД</li>
                        <li>ДЛЯ          ПОЛУЧЕНИЯ          ВОДИТЕЛЬСВОГО
УДОСТОВЕРЕНИЯ</li>
                    </ul>
                </div>
            </div>
        </div>
    </div>
</body>
</html>
*****

```

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

```
</body>
```

```
</html>
```

```
body {
```

```
  background: #fff;
```

```
  font-family: "Lato", Arial, Verdana;
```

```
}
```

```
h1, h2, h3, h4, h5, h6, .h1, .h2, .h3, .h4, .h5, .h6 {
```

```
  font-family: "Lato", Arial, Verdana;
```

```
  font-weight: 100;
```

```
  color: #444649;
```

```
}
```

```
* {
```

```
  margin: 0;
```

```
  padding: 0;
```

```
-moz-box-sizing: border-box;
```

```
  -o-box-sizing: border-box;
```

```
  -webkit-box-sizing: border-box;
```

```
  box-sizing: border-box;
```

```
}
```

```
ul {
```

```
  margin: 30px auto;
```

```
  text-align: center;
```

```
}
```

```
li {
```

```
  list-style: none;
```

```
    position: relative;
    display: inline-block;
    width: 100px;
    height: 100px;
}

@-moz-keyframes rotate {
    0% {transform: rotate(0deg);}
    100% {transform: rotate(-360deg);}
}

@-webkit-keyframes rotate {
    0% {transform: rotate(0deg);}
    100% {transform: rotate(-360deg);}
}

@-o-keyframes rotate {
    0% {transform: rotate(0deg);}
    100% {transform: rotate(-360deg);}
}

@keyframes rotate {
    0% {transform: rotate(0deg);}
    100% {transform: rotate(-360deg);}
}

.round {
    display: block;
    position: absolute;
    left: 0;
```



```
top: 0;
width: 100%;
height: 100%;
padding-top: 30px;
text-decoration: none;
text-align: center;
font-size: 25px;
text-shadow: 0 1px 0 rgba(255,255,255,.7);
letter-spacing: -.065em;
font-family: "Hammersmith One", sans-serif;
-webkit-transition: all .25s ease-in-out;
-o-transition: all .25s ease-in-out;
-moz-transition: all .25s ease-in-out;
transition: all .25s ease-in-out;
box-shadow: 2px 2px 7px rgba(0,0,0,.2);
border-radius: 300px;
z-index: 1;
border-width: 4px;
border-style: solid;
}

.round:hover {
width: 130%;
height: 130%;
left: -15%;
top: -15%;
font-size: 33px;
padding-top: 38px;
-webkit-box-shadow: 5px 5px 10px rgba(0,0,0,.3);
-o-box-shadow: 5px 5px 10px rgba(0,0,0,.3);
```

```
-moz-box-shadow: 5px 5px 10px rgba(0,0,0,.3);  
box-shadow: 5px 5px 10px rgba(0,0,0,.3);  
z-index: 2;  
border-size: 10px;  
-webkit-transform: rotate(-360deg);  
-moz-transform: rotate(-360deg);  
-o-transform: rotate(-360deg);  
transform: rotate(-360deg);  
}
```

Выпускная квалификационная работа выполнена мной совершенно самостоятельно. Все использованные в работе материалы и концепции из опубликованной научной литературы и других источников имеют ссылки на них.

«__» _____ г.

(подпись)

(Ф.И.О.)