

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«Национальный исследовательский ядерный университет «МИФИ»  
**Трехгорный технологический институт-**  
филиал федерального государственного автономного образовательного учреждения высшего образования  
«Национальный исследовательский ядерный университет «МИФИ»  
**(ТТИ НИЯУ МИФИ)**

**УТВЕРЖДАЮ**

  
Зам. директора  
Т.В. Труфанова  
«29» января 2025 г.

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО  
ПРОФЕССИОНАЛЬНОМУ МОДУЛЮ**

**ПМ.11 РАЗРАБОТКА, АДМИНИСТРИРОВАНИЕ И  
ЗАЩИТА БАЗ ДАННЫХ**

**Специальность:** 09.02.07. Информационные системы и программирование

**Квалификация выпускника:** администратор баз данных/ специалист по тестированию в области информационных технологий/программист/ технический писатель/ специалист по информационным системам/ специалист по информационным ресурсам/ разработчик веб и мультимедийных приложений

**Форма обучения:** очная

г. Трехгорный  
2025

## Содержание

1 Паспорт фонда оценочных средств .....	3
2 Результаты освоения профессионального модуля.....	7
3 Оценка освоения теоретического курса профессионального модуля .....	9

## 1 Паспорт фонда оценочных средств

Фонд оценочных средств (ФОС) предназначен для контроля и оценки знаний, полученных обучающимися за время освоения профессионального модуля «ПМ.11 Разработка, администрирование и защита баз данных».

ФОС включает контрольные материалы для проведения текущего контроля и промежуточной аттестации в форме экзамена.

ФОС разработан на основании следующих документов:

- Федерального государственного образовательного стандарта по специальности 09.02.07 «Информационные системы и программирование», утвержденного приказом Министерства просвещения Российской Федерации от 09 декабря 2016 г. № 1547;
- программы подготовки специалистов среднего звена по специальности СПО 09.02.07 «Информационные системы и программирование».

### 1.2. Результаты освоения профессионального модуля, подлежащие проверке

Перечень формируемых компетенций.

#### Перечень общих компетенций:

Код	Наименование общих компетенций
ОК 01.	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам.
ОК 02.	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности.
ОК 05.	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста.
ОК 09.	Пользоваться профессиональной документацией на государственном и иностранном языках.

#### Перечень профессиональных компетенций:

Код	Наименование видов деятельности и профессиональных компетенций
ПК 11.1	Осуществлять сбор, обработку и анализ информации для проектирования баз данных.
ПК 11.2.	Проектировать базу данных на основе анализа предметной области.

ПК 11.3.	Разрабатывать объекты базы данных в соответствии с результатами анализа предметной области.
ПК 11.4.	Реализовывать базу данных в конкретной системе управления базами данных.
ПК 11.5.	Администрировать базы данных.
ПК 11.6.	Защищать информацию в базе данных с использованием технологии защиты информации.

### Воспитательная работа

Профессиональный модуль специальность 09.02.07 «Информационные системы и программирование»		
Направление/ цели	Создание условий, обеспечивающих	Использование воспитательного потенциала учебной дисциплины
<b>Профессиональное воспитание</b>	- формирование чувства личной ответственности за научно-технологическое развитие России, за результаты исследований и их последствия <b>(B17)</b>	1. Использование воспитательного потенциала дисциплины профессионального модуля для формирования чувства личной ответственности за достижение лидерства России в ведущих научно-технических секторах и фундаментальных исследованиях, обеспечивающих ее экономическое развитие и внешнюю безопасность, посредством контекстного обучения, обсуждения социальной и практической значимости результатов научных исследований и технологических разработок. 2. Использование воспитательного потенциала дисциплины профессионального модуля для формирования социальной ответственности за результаты исследований и их последствия, развития исследовательских качеств посредством выполнения учебно-исследовательских заданий, ориентированных на изучение и проверку научных фактов, критический анализ публикаций в профессиональной области, вовлечения в реальные междисциплинарные научно-исследовательские проекты.
	- формирование ответственности за профессиональный выбор, профессиональное развитие и профессиональные решения <b>(B18)</b>	Использование воспитательного потенциала дисциплины для формирования у студентов ответственности за свое профессиональное развитие посредством выбора студентами индивидуальных образовательных траекторий, организации системы общения между всеми участниками образовательного процесса, в том числе с использованием новых информационных технологий.
	- формирование научного мировоззрения, культуры поиска нестандартных научно-технических решений, критического отношения к исследованиям лженаучного толка <b>(B19)</b>	1. Использование воспитательного потенциала дисциплины профессионального модуля для: - формирования понимания основных принципов и способов научного познания мира, развития исследовательских качеств студентов посредством их вовлечения в исследовательские проекты по областям научных исследований. 2. Использование воспитательного потенциала дисциплины для: - формирования способности отделять настоящие научные исследования от лженаучных посредством проведения со студентами занятий и регулярных

		<p>бесед; - формирования критического мышления, умения рассматривать различные исследования с экспертной позиции посредством обсуждения со студентами современных исследований, исторических предпосылок появления тех или иных открытий и теорий.</p>
	<p>- формирование навыков коммуникации, командной работы и лидерства <b>(B20)</b>;  - формирование способности и стремления следовать в профессии нормам поведения, обеспечивающим нравственный характер трудовой деятельности и неслужебного поведения <b>(B21)</b>;  - формирование творческого инженерного мышления, навыков организации коллективной проектной деятельности <b>(B22)</b></p>	<p>1.Использование воспитательного потенциала дисциплины профессионального модуля для развития навыков коммуникации, командной работы и лидерства, творческого инженерного мышления, стремления следовать в профессиональной деятельности нормам поведения, обеспечивающим нравственный характер трудовой деятельности и неслужебного поведения, ответственности за принятые решения через подготовку групповых курсовых работ и практических заданий, решение кейсов, прохождение практик и подготовку ВКР.  2.Использование воспитательного потенциала дисциплины для:  - формирования производственного коллективизма в ходе совместного решения как модельных, так и практических задач, а также путем подкрепление рационально-технологических навыков взаимодействия в проектной деятельности эмоциональным эффектом успешного взаимодействия, ощущением роста общей эффективности при распределении проектных задач в соответствии с сильными компетентными и эмоциональными свойствами членов проектной группы.</p>
	<p>- формирование культуры информационной безопасности <b>(B23)</b></p>	<p>Использование воспитательного потенциала дисциплины профессионального модуля для формирования базовых навыков информационной безопасности через изучение последствий халатного отношения к работе с информационными системами, базами данных (включая персональные данные), приемах и методах злоумышленников, потенциальном уровне пользователей.</p>
<p><b>Профессиональный модуль  по УГНС 09.00.00 «Информатика и вычислительная техника»</b></p>		
<p><b>Профессиональное воспитание</b></p>	<p>- формирование культуры решения изобретательских задач <b>(B26)</b>;  - формирование навыков цифровой гигиены <b>(B27)</b>;  - формирование ответственности за обеспечение кибербезопасности <b>(B28)</b>;  - формирование профессионально значимых установок: не производить, не копировать и не</p>	<p>1. Использование воспитательного потенциала дисциплины для формирования культуры написания и оформления программ, а также привития навыков командной работы за счет использования систем управления проектами и контроля версий.  2.Использование воспитательного потенциала дисциплины для формирования культуры решения изобретательских задач, развития логического мышления, путем погружения студентов в научную и инновационную деятельность института.  3.Использование воспитательного потенциала профильной дисциплины для формирования навыков цифровой гигиены, а также системности и гибкости мышления, посредством изучения методологических и технологических основ обеспечения</p>

	использовать программные и технические средства, не приобретённые на законных основаниях; не нарушать признанные нормы авторского права; не нарушать тайны передачи сообщений, не практиковать вскрытие информационных систем и сетей передачи данных; соблюдать конфиденциальность доверенной информации (В29)	информационной безопасности и кибербезопасности при выполнении и защите результатов учебных заданий и лабораторных работ по криптографическим методам защиты информации в компьютерных системах и сетях. 4.Использование воспитательного потенциала дисциплины для формирования культуры безопасного программирования посредством тематического акцентирования в содержании дисциплин и учебных заданий. 5.Использование воспитательного потенциала дисциплины для формирования системного подхода по обеспечению информационной безопасности и кибербезопасности в различных сферах деятельности посредством исследования и перенятия опыта постановки и решения научно-практических задач организациями-партнерами.
--	---	---

С целью овладения соответствующими общими компетенциями обучающийся в ходе освоения профессионального модуля должен **иметь практический опыт (О), знания (З) и умения (У).**

Результаты обучения: умения, знания	Осваиваемые компетенции
<b>Практический опыт:</b>	
О1. Работа с объектами базы данных в конкретной системе управления базами данных.	К 01.
О2. Использование стандартных методов защиты объектов базы данных.	ОК 02. ОК 05. ОК 09.
О3. Работа с документами отраслевой направленности.	ПК 11.1.
<b>Уметь:</b>	ПК 11.2.
У1. Работать с современными case-средствами проектирования баз данных.	ПК 11.3.
У2. Проектировать логическую и физическую схемы базы данных.	ПК 11.4.
У3. Создавать хранимые процедуры и триггеры на базах данных.	ПК 11.5.
У4. Применять стандартные методы для защиты объектов базы данных.	ПК 11.6.
У5. Выполнять стандартные процедуры резервного копирования и мониторинга выполнения этой процедуры.	
У6. Выполнять процедуру восстановления базы данных и вести мониторинг выполнения этой процедуры.	
У7. Обеспечивать информационную безопасность на уровне базы данных.	
<b>Знать:</b>	
З1. Основные положения теории баз данных, хранилищ данных, баз знаний.	
З2. Основные принципы структуризации и нормализации базы данных.	
З3. Основные принципы построения концептуальной, логической и физической модели данных.	

<p>34. Методы описания схем баз данных в современных системах управления базами данных.</p> <p>35. Структуры данных систем управления базами данных, общий подход к организации представлений, таблиц, индексов и кластеров.</p> <p>36. Методы организации целостности данных; способы контроля доступа к данным и управления привилегиями.</p> <p>37. Основные методы и средства защиты данных в базах данных.</p>	
---	--

## 2 Результаты освоения профессионального модуля

Текущий контроль по профессиональному модулю производится с использованием тестовых заданий и практических работ.

Критерии оценки тестовых заданий.

Процент выполнения задания:

- 90 % и более – отлично;
- От 75 до 89 % – хорошо;
- от 60 до 74 % – удовлетворительно;
- менее 60 % – неудовлетворительно.

Критерии оценки выполнения практических заданий.

Оценка 5 – «отлично» выставляется, если студент выполнил 100 % задания, демонстрирует знания теоретического и практического материала по теме практической работы, определяет взаимосвязи между показателями задания, дает правильный алгоритм выполнения поставленной задачи, самостоятельно делает необходимые выводы и обобщения по полученным результатам, дает четкие ответы на вопросы.

Оценка 4 – «хорошо» ставится, если студент выполнил не менее 75 % задания, демонстрирует знания теоретического и практического материала по теме практической работы, допуская незначительные неточности в алгоритме при выполнении задания, дает не совсем полный ответ на вопросы.

Оценка 3 – «удовлетворительно» ставится, если студент выполнил не менее 50 % задания, затрудняется с правильной оценкой предложенного задания, дает неполный ответ, требующий наводящих вопросов преподавателя, выбор алгоритма выполнения задания возможен при наводящих вопросах преподавателя.

Оценка 2 – «неудовлетворительно» ставится, если студент выполнил менее 50 % задания, дает неверную оценку ситуации, неправильно выбирает алгоритм действий, не дает правильный ответ на контрольные вопросы.

Промежуточной аттестацией по учебной дисциплине является экзамен.

К экзамену допускаются обучающиеся, успешно освоившие весь теоретический курс учебной дисциплины и выполнившие практические работы.

Итогом промежуточной аттестации по учебной дисциплине выступает оценка по пятибалльной шкале оценивания соответственно: «5» (отлично), «4» (хорошо), «3» (удовлетворительно), «2» (неудовлетворительно).

Экзамен проводится в устной форме.

Критерии оценки устного ответа студента.

При оценке устных ответов студентов учитываются следующие критерии:

1. Знание основных процессов изучаемой предметной области, глубина и полнота раскрытия вопроса.
2. Владение терминологическим аппаратом и использование его при ответе.
3. Умение объяснить сущность явлений, событий, процессов, делать выводы и обобщения, давать аргументированные ответы.
4. Владение монологической речью, логичность и последовательность ответа, умение отвечать на поставленные вопросы, выражать свое мнение по обсуждаемой проблеме.

Оценкой "ОТЛИЧНО" оценивается ответ, который показывает прочные знания основных процессов изучаемой предметной области, отличается глубиной и полнотой раскрытия темы; владение терминологическим аппаратом; умение объяснять сущность явлений, процессов, событий, делать выводы и обобщения, давать аргументированные ответы, приводить примеры; свободное владение монологической речью, логичность и последовательность ответа.

Оценкой "ХОРОШО" оценивается ответ, обнаруживающий прочные знания основных процессов изучаемой предметной области, отличается глубиной и полнотой раскрытия темы; владение терминологическим аппаратом; умение объяснять сущность явлений, процессов, событий, делать выводы и обобщения,

давать аргументированные ответы, приводить примеры; свободное владение монологической речью, логичность и последовательность ответа. Однако допускается одна - две неточности в ответе.

Оценкой "УДОВЛЕТВОРИТЕЛЬНО" оценивается ответ, свидетельствующий в основном о знании процессов изучаемой предметной области, отличающийся недостаточной глубиной и полнотой раскрытия темы; знанием основных вопросов теории; слабо сформированными навыками анализа явлений, процессов, недостаточным умением давать аргументированные ответы и приводить примеры; недостаточно свободным владением монологической речью, логичностью и последовательностью ответа. Допускается несколько ошибок в содержании ответа.

Оценкой "НЕУДОВЛЕТВОРИТЕЛЬНО" оценивается ответ, обнаруживающий незнание процессов изучаемой предметной области, отличающийся неглубоким раскрытием темы; незнанием основных вопросов теории, несформированными навыками анализа явлений, процессов; неумением давать аргументированные ответы, слабым владением монологической речью, отсутствием логичности и последовательности. Допускаются серьезные ошибки в содержании ответа.

### 3 Оценка освоения теоретического курса профессионального модуля

Структура фонда оценочных средств профессионального модуля «ПМ 11.

Разработка, администрирование и защита баз данных»

№ п/п	Контролируемые разделы, темы дисциплины	Формируемые компетенции	Вид аттестации	
			Текущий контроль	Промежуточная аттестация
1	МДК.11.01 Технология разработки, администрирования и защиты баз данных	ПК 11.1. ПК 12.2. ПК 13.3. ПК 14.4. ПК 15.4. ПК 16.4.	решение ситуационных задач, тестовые вопросы, создание и защита баз данных	дифференцированный зачет, контрольная работа
2	Учебная практика УП.11.01	ОК 01. ОК 02. ОК 05. ОК 09. У1-У7 31-37	анализ дневника ПП	Тестирование /устное собеседование характеристика с места прохождения практики оформление дневника по УП

№ п/п	Контролируемые разделы, темы дисциплины	Формируемые компетенции	Вид аттестации	
			Текущий контроль	Промежуточная аттестация
5	Производственная практика ПП.11.01		анализ дневника ПП	Тестирование /устное собеседование характеристика с места прохождения практики оформление дневника по ПП
6	Квалификационный экзамен по ПМ.11		тестовые вопросы,билеты	

### 3.1 Контрольно-оценочные средства

#### 3.1.1. Практические работы для подготовки к Демонстрационному экзамену

##### Задание 1 Разработка настольного приложения Баскетбольной системы управления

В рамках этого задания вы будете разрабатывать систему управления НБА.

##### Бизнес-описание проекта

Баскетбольная ассоциация «Фани» — городская коммерческая организация. Она часто проводит баскетбольные турниры среди любительских команд Города. Чтобы эффективнее работать с игроками и командами, а также для упрощения организации матчей, баскетбольная ассоциация «Фани» планирует разработать и внедрить Баскетбольную систему управления, которой смогут пользоваться поклонники этой игры. В системе предусмотрено четыре роли для пользователей: посетитель, игрок, менеджер команды и администратор.

Посетитель может посматривать краткое описание команд, прочесть историю их появления, узнать, где будут проводиться игры с их участием, форму каких цветов они носят и другую информацию. Также ему доступен просмотр всей информации и статистики матчей, в которых принимала участие каждая команда. Наконец, он может увидеть положение команд в турнирной таблице и личные данные игроков. Посетитель может зарегистрироваться в качестве игрока. При регистрации необходимо заполнить следующие поля: имя, пол, дата рождения, рост, вес, контакты

и позицию. После утверждения администратором посетитель становится игроком. После этого он может подать заявку на вступление в команду. Если менеджер команды утверждает его заявку, они подписывают контракт и оговаривают зарплату. Теперь после входа в систему и прохождения проверки он может просматривать расписание игр чемпионата и общаться с другими игроками с помощью коротких сообщений. Также он может бесплатно смотреть видеозаписи важных матчей в хорошем качестве. Администратор, пройдя проверку при входе в систему, может управлять информацией о командах и чемпионате, а также подтверждать регистрационную информацию игроков.

### **Пользователи**

С системой может работать большое количество разных типов пользователей:

- посетитель;
- администратор мероприятия;
- технический администратор.

### **Посетитель**

Посетитель может просматривать расписание игр, информацию об игроках и командах, статистику и другие сведения.

### **Администратор мероприятия**

Администратор мероприятия отвечает за составление расписания игр, работу с общей информацией об игровых сезонах, командах и игроках.

### **Технический администратор**

Технический администратор отвечает за составление технической статистики и просматривает отчеты.

Для своевременного окончания разработки системы к концу задания необходимо подготовить к сдаче описанные далее файлы и компоненты.

Проверьте, чтобы они соответствовали нижеописанным требованиям.

При создании всех компонентов системы вы должны использовать предоставленное руководство по стилю.

Убедитесь, что подготовлены все сообщения о проверке и ошибках для всех компонентов системы.

Убедитесь в работоспособности всех необходимых кнопок/гиперссылок в конце выполнения задания.

Убедитесь, что во всех компонентах системы используется единообразная система наименований.

В конце вы должны подготовить работоспособный проект и предоставить его для проверки.

### **3.1.2 Диаграмма сценариев использования**

В предоставленном материале описана работа Баскетбольной системы от Баскетбольной ассоциации «Фани». Проанализируйте работу системы и роли пользователей в соответствии с описанием работы системы и нарисуйте диаграмму.

В конце работы предоставьте файл с диаграммой сценариев использования с названием: use-case-diagram.vsdх.

## **1 Создайте информационную базу и импортируйте данные**

Вам необходимо создать информационную базу (базу данных) и импортировать в нее данные.

- Создайте информационную базу с именем «BasketballSystem».
- Необходимо создать таблицы Country (Страна), Player (Игрок), Team (Команда), PositionName (Название позиции), PositionOfPlayer (Позиция игрока) в соответствии со словарем данных (DataDictionary-BasketballSystem.docx), который предоставлен в материалах. При корреляции между таблицами должен использоваться уникальный серийный номер.

- Ключевые данные содержатся в файле data.xls и других дополнительных файлах. Некоторые из данных не могут быть напрямую импортированы в базу данных. Вам необходимо изменить их, чтобы они подходили для созданных вами таблиц. Все размещенные в архиве data.zip файлы данных, должны быть импортированы в информационную базу.

- Перед импортом таблицы Player проверьте, чтобы данные в поле Email (электронная почта) были корректными, а строка данных не повторялась.

- Перед импортом таблиц PositionName и PositionOfPlayer проверьте, чтобы для одного игрока была возможность указать несколько позиций.

## **2. Создание программного решения**

Создайте программное решение под именем «NBAManagement» на платформе .Net/ Java. В рамках следующих заданий вы будете создавать систему на базе этого программного решения.

## **3. Создание экрана MAIN SCREEN (Главный экран)**

Создайте этот экран интерфейса в соответствии с указаниями для экрана Main Screen в макете.

Это первый экран интерфейса, который появится при запуске приложения.

В нижней части этого экрана должны отображаться 3 фотографии с последних матчей. Создайте механизм, который при нажатии «<» или «>» будет перелистывать фотографии. При нажатии «>» должны появляться следующие 3 фотографии. При нажатии «<» должны появляться предыдущие 3 фотографии.

В нижней части каждого экрана/страницы приложения должна отображаться информация о текущем спортивном сезоне и о возрасте НБА. НБА была основана в 1946 году. Например, если сейчас идет спортивный сезон «2016–2017», на экране должно быть: «The current season is 2016–2017, and the NBA already has a history of 71 years».

Вы будете разрабатывать систему шаг за шагом, и некоторые функции будут добавляться в нее в рамках дальнейшей разработки.

## **4. Создание экрана VISITOR MAIN (Главный экран посетителя)**

Создайте этот экран интерфейса в соответствии с указаниями для экрана Visitor Main в макете.

На этом экране должны быть четыре кнопки:

- Teams (Команды)
- Players (Игроки)

- Matchups (Матчи)
- Photos (Фото)

Переход на этот экран осуществляется при нажатии на кнопку Visitor (Посетитель) на главном экране.

Вы будете разрабатывать систему шаг за шагом, и некоторые функции будут добавляться в нее в рамках дальнейшей разработки.

## **5. Создание экрана TEAMS MAIN (Главный экран команд)**

Создайте этот экран интерфейса в соответствии с указаниями для экрана Teams Main в макете.

На этом экране на разных вкладках должны отображаться команды Восточной Конференции и Западной конференции соответственно.

На вкладке Eastern (Восточная) должны отображаться команды Атлантического, Центрального и Юго-восточного дивизионов в соответствии с указаниями макета. На вкладке Western (Западная) должны отображаться команды Северо-западного, Тихоокеанского и Юго-западного дивизионов в соответствии с указаниями макета. Команды одного и того же дивизиона загружаются по названиям в алфавитном порядке.

Переход на этот экран осуществляется при нажатии на кнопку Teams (Команды) на главном экране посетителя.

## **6. Создание экрана TEAM DETAIL (Информация о команде)**

Создайте этот экран интерфейса в соответствии с указаниями для экрана TeamDetail в файле wireframe.ppsx.

На этом экране посетителям показывается подробная информация о команде, включая логотип команды, название, дивизион и конференцию. Следующая информация показывается на вкладках: Roster (полный состав), Matchup (матчи), First Lineup (стартовый состав).

На экране основном экране команд при нажатии на логотип или название команды открывается экран Team Detail. При переходе по ссылкам Roster, Matchup и

First Lineup открывается экран Team Detail, при этом должна быть выбрана соответствующая вкладка.

В поле Season (Сезон) отображается информация о конкретном сезоне из базы данных, отсортированная по времени в нисходящем порядке.

### **Вкладка Roster (полный состав)**

На этой вкладке представлен полный состав команды. Посетитель может осуществлять поиск по нему в соответствующем сезоне. Полный состав команды должен отображаться в виде сетки данных, включая:

- No.: Номер на форме
- Name: имя игрока
- Position: позиция игрока на поле
- Дата рождения
- College: колледж выпуска
- Experience: количество лет, проведенных в НБА — текущий год минус

выбранный год

- Зарплата

### **Вкладка Matchup (матчи)**

На этой вкладке отображается расписание игр команды в восходящем порядке в

соответствии со временем начала в сетке данных. Пользователь может производить поиск по расписанию игр команды в рамках сезона по разным параметрам, включая:

- Date: Дата игры
- Matchup Type: тип матча
- Opponent: соперник
- Starttime: время начала
- Result: результат в формате «Результат команды1 (Team1) - Результат команды2 (Team2)» либо «Результат гостевой команды (Team Away) - Результат команды хозяев (Team Home)»

- Location: место проведения
- Status: статус с параметрами Not Start (не начался), Running (идет) и Finished (завершен).

### **Вкладка Lineup (состав)**

На этой вкладке представлен состав команды на игру в виде диаграммы. Показывается пять позиций в соответствии с форматом, указанным в диаграмме. Вначале показываются игроки стартового состава на своих позициях, потом показываются другие игроки, которые могут выйти в игру на заменах.

Переход на этот экран осуществляется при нажатии на логотип команды на экране Teams Main.

## **Задание 2 Программная реализация функциональности системы управления НБА на базе задания 1.**

Перед тем, как вы начнете работу над этим заданием, вам необходимо завершить создание экранов Main Screen и Visitor Main, а также убедиться, что все ссылки работают корректно.

### **Бизнес-описание проекта**

Баскетбольная ассоциация «Фани» — городская коммерческая организация. Она часто проводит баскетбольные турниры среди любительских команд Города. Чтобы эффективнее работать с игроками и командами, а также для упрощения организации матчей, баскетбольная ассоциация «Фани» планирует разработать и внедрить Баскетбольную систему управления, которой смогут пользоваться поклонники этой игры. В системе предусмотрено четыре роли для пользователей: посетитель, игрок, менеджер команды и администратор.

Посетитель может посматривать краткое описание команд, прочесть историю их появления, узнать, где будут проводиться игры с их участием, форму каких цветов они носят и другую информацию. Также ему доступен просмотр всей информации и статистики матчей, в которых принимала участие каждая команда. Наконец, он может увидеть положение команд в турнирной таблице и личные данные игроков. Посетитель может зарегистрироваться в качестве игрока. При регистрации

необходимо заполнить следующие поля: имя, пол, дата рождения, рост, вес, контакты и позицию. После утверждения администратором посетитель становится игроком. После этого он может подать заявку на вступление в команду. Если менеджер команды утверждает его заявку, они подписывают контракт и оговаривают зарплату. Теперь после входа в систему и прохождения проверки он может просматривать расписание игр чемпионата и общаться с другими игроками с помощью коротких сообщений. Также он может бесплатно смотреть видеозаписи важных матчей в хорошем качестве. Администратор, пройдя проверку при входе в систему, может управлять информацией о командах и чемпионате, а также подтверждать регистрационную информацию игроков.

### **Пользователи**

С системой может работать большое количество разных типов пользователей:

- посетитель;
- администратор мероприятия;
- технический администратор.

### **Посетитель**

Посетитель может просматривать расписание игр, информацию об игроках и командах, статистику и другие сведения.

### **Администратор мероприятия**

Администратор мероприятия отвечает за составление расписания игр, работу с общей информацией об игровых сезонах, командах и игроках.

### **Технический администратор**

Технический администратор отвечает за составление технической статистики и просматривает отчеты.

Для своевременного окончания разработки системы к концу задания необходимо подготовить к сдаче описанные далее файлы и компоненты.

Проверьте, чтобы они соответствовали нижеописанным требованиям.

При создании всех компонентов системы вы должны использовать предоставленное руководство по стилю.

Убедитесь, что подготовлены все сообщения о проверке и ошибках для всех компонентов системы.

Убедитесь в работоспособности всех необходимых кнопок/гиперссылок в конце выполнения задания.

Убедитесь, что во всех компонентах системы используется единообразная система наименований.

В конце вы должны подготовить работоспособный проект и предоставить его для проверки.

### **3.1.3. Создание экрана PLAYERS MAIN (главный экран игроков)**

Создайте интерфейс этого экрана.

На этом экране посетители могут просматривать информацию об игроках. Пользователи могут осуществлять поиск игроков по сезонам, командам и именам игроков.

На этом экране должны отображаться 26 букв английского алфавита, от А до Z. Кликнув на соответствующую букву, посетитель может начать поиск нужной информации по инициалам имени игрока. Выбранная буква должна выделяться рамкой или иным способом.

В поле Season (Сезон) должна отображаться информация о сезонах из базы данных, отсортированная по времени в нисходящем порядке.

В поле Team (Команда) показываются все команды в системе в следующем виде: Team Name (Название команды).

Информация об игроках отсортирована по номеру на форме в восходящем порядке. В нее включены следующие сведения:

- Photo: фотография игрока
- No.: номер формы
- Name: имя игрока
- Team: команда игрока
- Postition: позиция игрока на площадке
- Weight

- Height
- Experience: количество лет, проведенных в НБА — текущий год минус выбранный год
- Country: гражданство

Необходимо разработать форму постраничного показа по 10 записей на каждой странице в виде списка. Страница должна иметь следующие возможности:

- Переход на первую страницу при нажатии.
- Переход на предыдущую страницу при нажатии.
- Переход на следующую страницу при нажатии.
- Переход на последнюю страницу при нажатии.
- На нужную страницу также можно перейти, введя номер конкретной страницы и нажав клавишу «Ввод» (Enter).

Общее количество записей и количество записей на каждой странице должно быть указано в нижней части экрана в следующем формате: «Total XX records, YY records in one page».

«Total XX records, YY records in one page».

Переход на этот экран осуществляется при клике на кнопку Players на экране Visitor Main.

#### **3.1.4. Создание экрана PLAYER DETAIL (информация об игроке)**

Создайте этот экран интерфейса.

На этом экране пользователям показывается информация об игроках, включая следующие поля: Photo (фото), No. (номер на форме), Height (рост), Team (команда), Date of Birth (дата рождения), Experience (опыт в НБА), College (колледж выпуска), Salary (зарплата); средняя результативность для каждой игры (PPG) в текущем сезоне, а также за время всей карьеры (Career), среднее количество голевых пасов для каждой игры (APG), среднее количество подборов для каждой игры (RPG). Расчет средних параметров для всей карьеры игрока (Career) производится без учета данных за текущий сезон.

На этой вкладке показана статистика каждого индекса игрока для каждой игры в текущем сезоне. Горизонтальная ось координатной сетки показывает дату игры, вертикальная ось — данные.

Посетители могут просматривать информацию об игроке, отсортированную по следующим типам данных:

- Points (очки)
- Rebounds (подборы)
- Assists (голевые пасы)
- Steals (отборы)
- Blocks (блок-шоты)

Когда посетитель нажимает на индекс, этот индекс подсвечивается голубым, а данные сразу загружаются и отображаются в виде ломаной линии. Среднее значение выбранного значения индекса отображается над ломаной линией.

Переход на этот экран осуществляется при нажатии на фото игрока на экране Players Main.

### **3.1.5. Создание экрана MATCHUP LIST (список матчей)**

Создайте интерфейс этого экрана. На этом экране посетители могут просматривать информацию обо всех матчах. Если на выбранную дату никаких матчей нет, то должно показываться соответствующее предупреждение.

Посетители должны иметь возможность выбрать дату, чтобы посмотреть информацию о матчах в этот день. Информация о матчах должна быть отсортирована по времени начала матча в восходящем порядке. Дата должна уменьшаться на один день при однократном нажатии «<»; дата должна увеличиваться на один день при однократном нажатии на «>».

Информация о матчах в конкретный день должна включать следующее:

- Статус: статус матча в первой колонке. Если игра еще не началась, показывать Not Start на голубом фоне; если игра уже идет, показывать Running на красном фоне; если игра завершилась, показывать Finished на сером фоне;

- Логотип гостевой команды;
- Название гостевой команды;
- Результат: формат этой колонки должен быть «Результат гостевой команды – Результат команды хозяев». Если игра еще не началась, показывать «—»; если игра продолжается, показывать количество очков в реальном времени; если игра завершена, показывать окончательный результат;
- Логотип команды хозяев;
- Название команды хозяев;
- Место проведения.

Над интерфейсом экрана должна отображаться игра, начало которой ближе всего к текущему времени, и следующая информация: логотипы команды гостей и команды хозяев, названия команд и время начала игры. Если все игры на текущей день уже завершились, то отображается следующая информация о последнем матче за день: логотипы команды гостей и команды хозяев, названия команд, счет.

Подробности матчей показываются при нажатии кнопки View (показать). Если игра еще не началась, то кнопка View должна быть неактивна.

Подробности относительно работы кнопки View описаны в п. 2.4 Matchup Detail.

Переход на этот экран осуществляется при нажатии на кнопку Matchups на экране Visitor Main.

### **3.1.6. Создание экрана MATCHUP DETAIL (информация о матче)**

Создайте этот экран интерфейса. На этом экране посетители могут просматривать подробную информацию о матчах.

Над формой должна отображаться информация о командах и результаты. Информация о командах должна включать название команды и логотип. Результаты включают общий итог и количество очков за каждую четверть. Название и очки выигравшей команды должны отображаться жирным шрифтом. Если игра переходит в овертайм, то необходимо загружать также данные и по овертайму, с указанием

«ОТ1» для первого овертайма, «ОТ2» для второго и т. д. Если матч еще не закончен, то очки для тех четвертей, которые еще не начались, должны отображаться как «».

Статус: под результатом должен отображаться статус матча. Если игра еще не началась, Not Start на голубом фоне; если игра уже идет, показывать Running на красном фоне; если игра завершилась, показывать Finished на сером фоне.

Если игра еще не завершена, то в эти поля необходимо загружать данные текущего положения на поле.

### **Вкладка Team Status (статус команды)**

На этой вкладке сравниваются данные о матчах обеих команд в виде списка, в котором указывается следующее:

- FG Made-Attempted – это поле имеет формат «Количество результативных бросков с игры – количество нерезультативных бросков»;
- 3PT Made-Attempted – это поле имеет формат «Количество результативных 3-очковых бросков – количество нерезультативных 3-очковых бросков»;
- FT Made-Attempted – это поле имеет формат «Количество результативных штрафных бросков – количество нерезультативных штрафных бросков»;
- Подборы;
- Голевые пасы;
- Отборы;
- Блок-шоты;
- Потери.

Над списком необходимо загрузить логотипы обеих команд, логотип гостевой команды должен быть на первом плане, команды хозяев — на втором.

Правее списка указать, соответственно, % результативных бросков и % результативных трехочковых бросков для обеих команд в виде гистограммы (графика со столбцами). Слева должны показываться сокращенные названия команд, а справа — соответствующие % результативных бросков и трехочковых бросков.

- **Field Goal%:** процент результативных бросков, который является отношением результативных бросков к общему количеству бросков.
- **Three Point%:** процент результативных трехочковых бросков, который является отношением результативных трехочковых бросков к общему количеству трехочковых бросков.

### **Вкладка Short Chart (Краткая таблица)**

В середине этой вкладки должен загружаться логотип команды хозяев. По бокам загрузите логотипы обеих команд и информацию об игроках стартового состава. Слева выведите информацию об игроках гостевой команды, а справа — домашней команды, включая следующее:

- Фотографии
- Имя: формат этого поля — «Имя (номер на форме)».

### **Вкладка Log (Журнал)**

На этой вкладке должны подробно отображаться все события каждой четверти в виде списка, указанные события должны быть отсортированы по времени в нисходящем порядке, включая следующие поля:

- **OccurTime:** время события
- **Team:** название команды (сокращ.)
- **Player:** формат поля — «Имя игрока (Номер на форме)»
- **ActionType:** тип действия
- **Remark:** комментарий

Посетители могут выбрать нужную четверть, чтобы увидеть вышеуказанные подробности матча. Если игра переходит в овертайм, загрузите также данные и по овертайму, с указанием «OT1» для первого овертайма, «OT2» для второго и т. д.

Переход на этот экран осуществляется при нажатии на кнопку View (показать) на экране Matchups List.

### **3.1.7. Создать экран PHOTOS (ФОТОГРАФИИ)**

Создайте интерфейс этого экрана. На этом экране посетители могут увидеть лучшие фото.

Фотографии должны отображаться постранично и с сортировкой по времени создания в нисходящем порядке. На экране интерфейса должны показываться превью фотографий.

При щелчке правой кнопкой мыши по фотографии должно появляться меню опций, как указано в диаграмме. Также необходимо реализовать возможность скачивания одной фотографии.

При нажатии на кнопку `Download pictures in current page` (Скачать все фотографии на текущей странице), все фотографии с текущей страницы должны скачиваться в указанное место локального диска. После завершения скачивания укажите пользователю место скачивания.

Создайте функцию пролистывания страниц. На каждой странице должно отображаться по 12 фотографий. Пролистывание должно состоять из следующих функций:

- Переход на первую страницу при нажатии «|<»;
- Переход на предыдущую страницу при нажатии «<»;
- Переход на следующую страницу при нажатии «>»
- Переход на последнюю страницу при нажатии «>|».
- Должна быть возможность ввода конкретного номера страницы, при нажатии клавиши Ввод (Enter) осуществляется переход на указанную страницу.

В нижней части экрана должно быть указано общее количество фотографий, количество фотографий на каждой странице и общее количество страниц в следующем формате: «Total XX pictures, YY pictures in one page, Total ZZ Pages».

Переход на этот экран осуществляется при нажатии на кнопку `Photos` на экране `Visitor Main`.

### **3.1.8. Построение базы данных**

Вам необходимо спланировать строение базы данных Системы управления НБА для баскетбольной ассоциации «Фани». Создавать эту систему не требуется. Вам

нужно только предоставить диаграмму сущностей и связей (ERD-Database Diagram) и словарь данных (Data Dictionary).

Требования баскетбольной ассоциации "Фани" к построению базы данных изложены в материалах. Внимательно изучите их, выделите нужные сущности и проанализируйте их связи. Изучите предоставленные в материалах шаблон ERD (ERD-template.vsdх) и словарь данных (DataDictionary-template.xlsx), после чего составьте ERD и соответствующий словарь данных. В конце работы создайте файл с построенной базой данных ERD-final.vsdх и словарь данных "DataDictionary-final.xlsx".

### **Задание 3 Программная реализация функциональности системы управления НБА на базе задания 2.**

В рамках этого задания вы будете реализовывать функциональность системы управления НБА на базе своей предыдущей работы (задания 2). Перед началом работы над этим заданием, вам необходимо завершить создание экрана Main Screen, а также убедиться, что все ссылки работают корректно.

#### **Бизнес-описание проекта**

Баскетбольная ассоциация «Фани» — городская коммерческая организация. Она часто проводит баскетбольные турниры среди любительских команд Города. Чтобы эффективнее работать с игроками и командами, а также для упрощения организации матчей, баскетбольная ассоциация «Фани» планирует разработать и внедрить Баскетбольную систему управления, которой смогут пользоваться поклонники этой игры. В системе предусмотрено четыре роли для пользователей: посетитель, игрок, менеджер команды и администратор.

Посетитель может просматривать краткое описание команд, прочесть историю их появления, узнать, где будут проводиться игры с их участием, форму каких цветов они носят и другую информацию. Также ему доступен просмотр всей информации и статистики матчей, в которых принимала участие каждая команда. Наконец, он может увидеть положение команд в турнирной таблице и личные данные игроков. Посетитель может зарегистрироваться в качестве игрока. При регистрации

необходимо заполнить следующие поля: имя, пол, дата рождения, рост, вес, контакты и позицию. После утверждения администратором посетитель становится игроком. После этого он может подать заявку на вступление в команду. Если менеджер команды утверждает его заявку, они подписывают контракт и оговаривают зарплату. Теперь после входа в систему и прохождения проверки он может просматривать расписание игр чемпионата и общаться с другими игроками с помощью коротких сообщений. Также он может бесплатно смотреть видеозаписи важных матчей в хорошем качестве. Администратор, пройдя проверку при входе в систему, может управлять информацией о командах и чемпионате, а также подтверждать регистрационную информацию игроков.

### **Пользователи**

С системой может работать большое количество разных типов пользователей:

- посетитель;
- администратор мероприятия;
- технический администратор.

### **Посетитель**

Посетитель может просматривать расписание игр, информацию об игроках и командах, статистику и другие сведения.

### **Администратор мероприятия**

Администратор мероприятия отвечает за составление расписания игр, работу с общей информацией об игровых сезонах, командах и игроках.

### **Технический администратор**

Технический администратор отвечает за составление технической статистики и просматривает отчеты.

Для своевременного окончания разработки системы к концу задания необходимо подготовить к сдаче описанные далее файлы и компоненты.

Проверьте, чтобы они соответствовали нижеописанным требованиям.

При создании всех компонентов системы вы должны использовать предоставленное руководство по стилю.

Убедитесь, что подготовлены все сообщения о проверке и ошибках для всех компонентов системы.

Убедитесь в работоспособности всех необходимых кнопок/гиперссылок в конце выполнения задания.

Убедитесь, что во всех компонентах системы используется единообразная система наименований.

В конце вы должны подготовить работоспособный проект и предоставить его для проверки.

### **3.1.9. Создание экрана ADMIN LOGIN (Вход для администратора)**

Необходимо разработать и создать механизм авторизации пользователей в системе.

После успешной авторизации администратор может войти в систему.

Для успешного входа администратор должен ввести номер сотрудника и пароль; учетные данные должны быть сверены с данными в приложении.

При неудачной попытке входа должно показываться соответствующее уведомление, и пользователь в систему попадать не должен.

Существует два типа администраторов:

- Event Administrator (администратор мероприятия)
- Technical Administrator (технический администратор)

После успешного входа администратора должен осуществляться переход на соответствующий главный экран в соответствии с их рабочей функцией:

- Event Administrator (администратор мероприятия)
- Technical Administrator (технический администратор)

Нужно предусмотреть автоматическое запоминание последнего успешно вошедшего в систему пользователя.

Нужно предусмотреть механизм быстрого завершения сеанса (выхода пользователя из системы) - например, в правом верхнем углу должна быть кнопка Logout.

### **3.1.10. Создание экрана EVENT ADMINISTRATOR MENU (меню администратора мероприятия)**

Создайте интерфейс этого экрана. На этом экране должны быть расположены 4 кнопки:

- Manage Seasons (управление сезонами)
- Manage Matchups (управление матчами)
- Manage Teams (управление командами)
- Manage Players (управление игроками)

Вы будете разрабатывать систему постепенно, и некоторые функции будут добавляться в нее шаг за шагом при дальнейшей разработке.

### **3.1.11. Создание экрана TECHNICAL ADMINISTRATOR MENU (меню технического администратора)**

Создайте интерфейс этого экрана. На этом экране должны быть расположены 2 кнопки:

- Manage Executions (управление назначениями)
- Team Report (отчет о команде)

Вы будете разрабатывать систему постепенно, и некоторые функции будут добавляться в нее шаг за шагом при дальнейшей разработке.

### **3.1.12. Создание экрана MANAGE SEASONS (управление сезонами)**

Создайте этот экран интерфейса. На этом экране администратор может работать с информацией об игровом сезоне.

Администратор может отфильтровывать информацию по сезонам и типам матчей.

В поле Season (сезон) должна отображаться информация о сезоне из базы данных, отсортированная по времени в нисходящем порядке.

Поле «Тип матча» включает следующие значения:

- ALL (все матчи)
- Preseason (до начала сезона)

- Regular Season (регулярный сезон)
- Post Season (после сезона)

Загрузите информацию о сезоне в сетку данных слева, включая следующее:

- Season (сезон)
- Matchup Type (тип матча)
- Number of Matchup (номер матча)

При выборе ряда записей в сетке данных слева, в сетку данных справа должна загружаться информация обо всех матчах сезона, отсортированная по времени начала матчей в восходящем порядке, включая следующее:

- Дата
- Матч: названия двух противостоящих команд в следующем формате: «Гостевая команда @ Команда хозяев»

- Результат в формате «Очки гостевой команды – очки команды хозяев».
- Место проведения

Переход на этот экран осуществляется по нажатию кнопки Manage Seasons на экране Event Administrator Menu.

### **3.1.13. Создание экрана MANAGE MATCHUPS (управление матчами)**

Создайте этот экран интерфейса. На этом экране администратор может работать с информацией о матчах, куда входят матчи до начала сезона, матчи регулярного чемпионата и матчи после окончания сезона.

Администратор может отфильтровывать информацию по сезонам и датам матчей.

В поле Season (сезон) должна отображаться информация о сезоне из базы данных, отсортированная по времени в нисходящем порядке.

При установке флажка рядом с полем Match Date (дата матча) статус поля «Дата матча» становится активным, и в сетке показываются только те матчи, которые будут проводиться в этот день. Если этот флажок не установлен, то поле даты матча неактивно, и дата матча не учитывается при поиске данных.

На вкладке Pre-Season Game (игры до начала сезона) в сетке данных должна отображаться информация о матчах, отсортированных по времени начала в восходящем порядке, включая следующую информацию:

- Дата
- Команда (гости)
- Команда (хозяева)
- Время
- Место проведения
- Завершено

Матч может быть удален нажатием на кнопку Delete (удалить) в строке записей. Прошедшие матчи, а также матчи, которые уже начались, не могут быть удалены, при попытке удаления должно показываться соответствующее уведомление.

Примечание: функцию кнопки Update (обновить) реализовывать не надо. При нажатии кнопки Update в строке записей должно отображаться сообщение, что эта функция не реализована.

На вкладке Regular Game (игры регулярного чемпионата) в сетке данных должна отображаться информация о матчах, отсортированных по времени начала в восходящем порядке, включая следующую информацию:

- Дата
- Команда (гости)
- Команда (хозяева)
- Время
- Место проведения
- Завершено

Матч может быть удален нажатием на кнопку Delete (удалить) в строке записей. Прошедшие матчи, а также матчи, которые уже начались, не могут быть удалены, при попытке удаления должно показываться соответствующее уведомление.

Примечание: функцию кнопки Update (обновить) реализовывать не надо. При нажатии кнопки Update в строке записей должно отображаться сообщение, что эта функция не реализована.

При нажатии на кнопку Export to Excel (экспорт в Excel) должен проводиться экспорт информации о матчах регулярного чемпионата из сетки данных в файл формата Excel.

Для получения дополнительной информации о кнопке «+ Add a new matchup for regular season» (добавить новый матч) см. раздел Результаты 3.6 Добавление нового матча в регулярный чемпионат.

Переход на этот экран осуществляется по нажатию кнопки Manage Matchups на экране Event Administrator Menu.

### **3.1.14. Создание экрана ADD A NEW MATCHUP FOR REGULAR SEASON (добавление нового матча в регулярный чемпионат)**

Создайте этот экран интерфейса. На этом экране администратор мероприятия может добавлять матчи в регулярный чемпионат сезона. Информация о добавленных матчах включает в себя следующее:

- Сезон
- Тип матча
- Дата
- Время
- Место проведения
- Команда (гости)
- Команда (хозяева)

Команды гостей и хозяев не могут быть совпадать. При выборе команды хозяев местом проведения матча по умолчанию будет считаться место расположения этой команды. Администратор может также самостоятельно ввести место проведения встречи.

Сохраните добавленные матчи в базу данных, нажав кнопку Submit (отправить).

Переход на этот экран осуществляется при нажатии на кнопку Add a new matchup (+ добавить новый матч) на вкладке Regular Season экрана Manage Matchups.

### **3.1.15. Создание экрана MANAGE TEAMS (управление командами)**

Создайте этот экран интерфейса. На этом экране администратор мероприятия может работать с данными о командах текущего сезона.

Администратор может отфильтровывать информацию по конференции, дивизиону и названию команды.

Содержание поля Conference (конференция) может принимать значение All (все) и названия обеих конференций. Содержание поля Division (дивизион) может принимать значение All (все) и названия всех дивизионов. К текстовому полю Team Name применяется поиск по сходству.

Информация о команде включает в себя следующее:

- Логотип команды
- Название
- Конференция
- Дивизион
- Тренер
- TotalSalary: сумма зарплат всех игроков команды в текущем сезоне в млн. долл. США

Переход на этот экран осуществляется по нажатию кнопки Manage Teams на экране Event Administrator Menu.

### **3.1.16. Создание экрана MANAGE PLAYERS (управление игроками)**

Создайте этот экран интерфейса. На этом экране администратор мероприятия может работать с информацией об игроках.

Администратор может фильтровать данные по полям Player Name (имя игрока), Position (позиция) и Country (гражданство). Поиск по имени игрока работает по сходству.

Поле Position может принимать значения All (все) и все позиции в таблицах; All означает выбор всех позиций.

Поле Country (гражданство) может принимать значения All (все) и все названия стран; All означает выбор всех стран.

Информация об игроке включает следующее:

- Имя
- Позиция
- Год прихода в команду
- Рост
- Вес
- Дата рождения
- Колледж выпуска
- Гражданство

Переход на этот экран осуществляется по нажатию кнопки Manage Players на экране Event Administrator Menu.

### **3.1.17. Создание экрана TEAM REPORT (отчет о команде)**

Создайте этот экран интерфейса. На этом экране технический администратор может распечатать отчет по командам текущего сезона. Технический администратор может выбирать просмотр либо Average (усредненных данных), либо Total (всех данных) для загрузки контента, а также может производить фильтрацию по типам матчей и сортировку по техническим данным.

Поле Matchup Type (тип матча) может принимать следующие значения:

- Preseason (до начала сезона)
- Regular Season (регулярный сезон)
- Post Season (после сезона)

Поле Rank (положение) содержит следующую информацию:

- Очки
- Подборы
- Голевые пасы
- Отборы
- Блок-шоты

Поле View (вид) содержит следующую информацию:

- В среднем
- Всего

Содержание информации о команде в отчете включает следующее:

- Положение
- Название
- Конференция
- Дивизион
- Очки
- Подборы
- Голевые пасы
- Отборы
- Блок-шоты
- Потери
- Фолы

Если технические данные в индексе команд оказываются одинаковыми, то рассчитайте положения команд в соответствии с Таблицей 1, приведенной ниже.

Таблица 1. Правила расчета положения для одинаковых технических данных

Технические данные	Положение
53	#1
53	#1
52	#2
50	#3

Содержимое отчета отображается постранично по 20 записей на каждой странице.

Переход на этот экран осуществляется по нажатию кнопки Team Report на экране Technical Administrator Menu.

#### **Задание 4 Разработка настольного приложения системы управления Национальным чемпионатом**

Эта система будет работать как общественная система (например, позволяя людям, узнать о движении чемпионата, о месте проведения) и закрытой части (например, участникам просматривать результаты и баллы, координатору управлять волонтерами и спонсорами и т.д.).

Система предназначена для различных типов пользователей:

- Внешние пользователи (не зарегистрировались в системе);
- Участники;
- Координаторы;
- Администраторы;
- Эксперты.

##### **Участник**

Когда Участник зарегистрировался в системе, он может изменить свой пароль и просмотреть свои результаты; Просматривать информацию о соответствующей компетенции в чемпионате, например конкурсанты, эксперты, план застройки, инфраструктурный лист и график и т. д.

##### **Координатор**

Координаторы в основном координируют волонтеров и спонсоров. Они могут управлять волонтерами, импортировать информацию о волонтерах и корректировать навыки, волонтеров; Получать статистику спонсорства и экспортировать детали спонсорства.

##### **Администратор**

Администраторы управляют участниками, информацией о событиях и информацией о компетенциях в системе.

## Эксперт

Эксперты могут просматривать информацию о компетенции, которую они оценивают, например. Участники, оценка и т.д. Главный эксперт может определять рабочие места конкурентов путем жеребьевки и вводить оценки.

В процессе выполнения данного задания вами должны быть достигнуты все практические результаты, необходимые для того, чтобы заказчик был спокоен, что система будет завершена вовремя.

Убедитесь, что вы следуете предоставленному руководству по стилю во всех частях системы.

Убедитесь, что вы предоставляете соответствующие проверки и сообщения об ошибках во всех частях системы.

Убедитесь, что все соответствующие кнопки / ссылки работают.

Убедитесь, что вы используете соответствующие соглашения об именах для всех частей системы по мере необходимости.

### **4.1 Разработка ERD диаграммы**

Разработайте ERD-диаграмму и словарь данных, которые бы обеспечивали нормальное функционирование системы и возможность хранения следующей информации:

- Компетенции (включая блоки, описание, расписание, план застройки и инфраструктуру);
- Чемпионаты (включая прошедшие и текущий, а также все результаты каждого участника);
- Пользователи (включая участников, экспертов, координаторов, администраторов и их фотографии);
- Волонтеры (включая распределение по компетенциям).

### **4.2 Создание базы данных и загрузка данных**

В соответствии с ERD-диаграммой создайте базу данных. Выполните импорт данных в базу данных.

Эти данные не отформатированы для импортирования непосредственно в базу данных, вам необходимо отформатировать данные и загрузить их в таблицы, которые вы только что создали.

Обратите внимание, что для каждого участника должен автоматически генерироваться IdNumber, который будет уникальным образом идентифицировать участника. IdNumber должен состоять из 15 символов в следующем формате:

*Год рождения (4 цифры) + Номер компетенции (2 цифры) + Порядковый номер (8 цифр) + Код клиентской группы (1 буква)*

Порядковый номер должен быть сквозным по всем годам в рамках каждой компетенции.

Для российских компетенций в номер компетенции добавляется префикс «R».

### **4.3 Создание приложения**

Создайте приложение, используя выбранную вами платформу.

### **4.4 Создание «Главный экран системы»**

Создайте главное меню системы.

Через эту форму незарегистрированные пользователи смогут узнать о чемпионате.

Вы будете разрабатывать систему и некоторые функции будут постепенно добавляться в разработку.

Каждое окно / страница приложения, отображает информацию:

*«? дней ? часов и ? минут до начала Национального чемпионата»*

в нижней части экрана, которая должна автоматически обновляться в режиме реального времени. Рассчитать количество времени, оставшегося до начала Национального чемпионата (начинается 2017-05-16 9:00).

### **4.5 Создание «Авторизация»**

Создайте форму. Эта форма позволяет пользователю войти в систему. Пользователи должны ввести их IdNumber и пароль которые будут проверяться по

базе данных, а также код аутентификации, который сверяется с зашумленным изображением (по технологии Captcha). Должны выводиться соответствующие сообщения, если вход в систему не увенчался успехом. Всего будет четыре различных типа пользователей для этой системы: конкурсант, координатор, администратор и эксперт.

Код аутентификации находится в формате изображения с шумом и 4 символами. Изображение аутентификации должно иметь белый фон, черный шрифт и серый шум. Требования следующие:

- Не менее 1 прописной буквы;
- Не менее 1 строчной буквы;
- Не менее 1 цифры;
- Код аутентификации и шум будет изменен, если щелкнуть по его области картинки;
- Код аутентификации и шум будет обновляться автоматически, если происходит сбой авторизации.

При установленном флажке "Запомнить меня", IdNumber и пароль должны будут автоматически загружаться при повторном запуске приложения.

Если авторизация неудачна в течение трех раз подряд, то система автоматически очистит поля ввода (IdNumber, Пароль, Код аутентификации) и вход в систему будет запрещен. Снятие блокировки на вход в систему только поле перезапуска приложения.

После успешного входа в систему должно отразиться меню, соответствующее роли пользователя:

- Участник (конкурсант): «Меню участника»;
- Координатор: «Меню координатора»;
- Администратор: «Меню администратора»;
- Эксперт: «Меню эксперта».

Когда пользователь нажимает кнопку «Выход» в любом окне / форме приложения, он выходит из системы и возвращается к «1. Главный экран».

Если для какого-либо не реализованного окна / формы приложения предусмотрен переход, то при его срабатывании должно выводиться сообщение:

*«находится в стадии разработки»*

#### **4.6 Создание «Меню участника»**

Создайте форму « Меню участника». Эта форма состоит из приветственной информации и 3 кнопок.

Требования к приветственной информации:

Отображать: «Доброе утро / Добрый день / Добрый вечер» в соответствии с текущим системным временем:

- Утро (06:00:00-11:59:59);
- День (12:00:00 – 17:59:59);
- Вечер (18:00:00 – 05:59:59).

Отображать: «Mrs./Mr. Имя пользователя».

Отображать: фотографию пользователя.

При дальнейшей разработке системы данное меню дополнится.

#### **4.7 Создание «Меню координатора»**

Создайте форму «Меню координатора». Эта форма состоит и приветственной информации и 3 кнопок.

Требования к приветственной информации:

Отображать: «Доброе утро / Добрый день / Добрый вечер» в соответствии с текущим системным временем:

- Утро (06:00:00-11:59:59);
- День (12:00:00 – 17:59:59);
- Вечер (18:00:00 – 05:59:59).

Отображать: «Mrs./Mr. Имя пользователя».

Отображать: фотографию пользователя.

При дальнейшей разработке системы данное меню дополнится.

#### **4.8 Создание «Меню администратора»**

Создайте форму «Меню администратора». Эта форма состоит и приветственной информации и 2 кнопок.

Требования к приветственной информации:

Отображать: «Доброе утро / Добрый день / Добрый вечер» в соответствии с текущим системным временем:

- Утро (06:00:00-11:59:59);
- День (12:00:00 – 17:59:59);
- Вечер (18:00:00 – 05:59:59).

Отображать: «Mrs./Mr. Имя пользователя».

Отображать: фотографию пользователя.

При дальнейшей разработке системы данное меню дополнится.

#### **4.9 Создание «Меню эксперта»**

Создайте форму «Меню эксперта». Эта форма состоит и приветственной информации и 3 кнопок.

Требования к приветственной информации:

Отображать: «Доброе утро / Добрый день / Добрый вечер» в соответствии с текущим системным временем:

- Утро (06:00:00-11:59:59);
- День (12:00:00 – 17:59:59);
- Вечер (18:00:00 – 05:59:59).

Отображать: «Mrs./Mr. Имя пользователя».

Отображать: фотографию пользователя.

Если эксперт зарегистрирован в системе, но на данном чемпионате не участвует в оценке – эти три кнопки должны быть неактивны. В таком случае при попытке нажать на любую из кнопок, должно выводиться соответствующее сообщение.

При дальнейшей разработке системы данное меню дополнится.

#### 4.10 Создание «Мой профиль»

Создайте форму «Мой профиль». Эта форма позволяет участникам просматривать свой профиль, включая IdNumber, имя, пол, регион, e-mail, и т.д. Участники (вошедшие в систему) могут изменять свой пароль.

Любая другая информация не может быть изменена участником, так как изменения этих данных может производить только администратор. Описание:

- Имя
- Пол
- Дата рождения
- IdNumber
- Регион. В формате (номер региона - наименование)
- Телефон
- E-mail

Если флажок «Смена пароля» не установлен, флажки «Пароль», «Повтор пароля» и «Видимый пароль», а также кнопка «Save» не активны.

Если установлен флажок «Смена пароля», будут активны поля «Пароль», «Повтор пароля» и «Видимый пароль», а также кнопка «Save», и будет включена функция изменения пароля.

При первом открытии этой формы флажок «Изменить пароль» по умолчанию не выбран.

Пароль должен соответствовать следующим требованиям:

- Не менее 6 символов и не более 16 символов
- Не менее 1 прописной буквы
- Не менее 1 строчной буквы
- Не менее 1 цифры
- Не содержит три одинаковых символа подряд
- Не содержит пробел

При изменении пароля значение «Повтор пароля» должно совпадать со значением «Пароль».

Если установлен флажок «Видимый пароль», то в поле «Пароль» отображаются символы пароля. Если флажок «Видимый пароль» не установлен, то в поле «Пароль» отображаются маскирующие символы.

При нажатии кнопки «Save» должна производиться запись изменений в базу данных.

#### **4.11 Создание «Моя компетенция»**

Создайте форму «Моя компетенция». Эта форма показывает участнику информацию о компетенции, в которой он / она участвует в текущем соревновании, включая участников, экспертов, план застройки, инфраструктуру и расписание. Если участник (который в настоящее время вошел в систему) не зарегистрирован на участие в текущем соревновании, необходимо вывести соответствующее сообщение.

Отображать информацию о компетенции, в которой участвует конкурсант в виде: «Номер компетенции – Название Компетенции».

##### **Вкладка «Участники»**

Показывается информация обо всех участниках, включенных в компетенцию, которую участник посещает в форме изображения и текста, включая фотографию, имя и регион участника.

Стиль отображения информации об участниках может быть изменен в соответствии с контекстным меню, которое включает в себя: большие изображения, маленькие изображения.

##### **Вкладка «Эксперты»**

Показывается вся информация о экспертах, включенных в компетенцию, которую участник посещает в форме изображения и текста, включая фотографию, имя и регион эксперта.

##### **Вкладка «План застройки»**

Показывается изображение плана застройки компетенции в соответствии с компетенцией. участника. Если плана застройки компетенции не

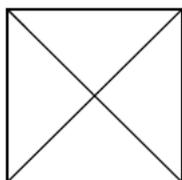
##### **Вкладка «Инфраструктура»**

Показывается инфраструктурный лист компетенции, в которой участник будет участвовать, также в виде рисунка.

### **Вкладка «Расписание»**

Показывается в виде текста график работы компетенции, в которой участвует конкурсант.

Если одна из вкладок не заполнена релевантной информацией – то необходимо вывести соответствующее сообщение или знак:



## **Задание 5 Разработка настольного приложения системы управления национальным чемпионатом на базе задания 4**

Эта система будет работать как общественная система (например, позволяя людям, узнать о движении чемпионата, о месте проведения) и закрытой части (например, участникам просматривать результаты и баллы, координатору управлять волонтерами и спонсорами и т.д.).

Система предназначена для различных типов пользователей:

- Внешние пользователи (не зарегистрировались в системе);
- Участники;
- Координаторы;
- Администраторы;
- Эксперты.

### **Участник**

Когда Участник зарегистрировался в системе, он может изменить свой пароль и просмотреть свои результаты; Просматривать информацию о соответствующей компетенции в чемпионате, например конкурсанты, эксперты, план застройки, инфраструктурный лист и график и т. д.

### **Координатор**

Координаторы в основном координируют волонтеров и спонсоров. Они могут управлять волонтерами, импортировать информацию о волонтерах и корректировать навыки, волонтеров; Получать статистику спонсорства и экспортировать детали спонсорства.

#### Администратор

Администраторы управляют участниками, информацией о событиях и информацией о компетенциях в системе.

#### Эксперт

Эксперты могут просматривать информацию о компетенции, которую они оценивают, например. Участники, оценка и т.д. Главный эксперт может определять рабочие места конкурентов путем жеребьевки и вводить оценки.

В процессе выполнения данного задания вами должны быть достигнуты все практические результаты, необходимые для того, чтобы заказчик был спокоен, что система будет завершена вовремя.

Убедитесь, что вы следуете предоставленному руководству по стилю во всех частях системы.

Убедитесь, что вы предоставляете соответствующие проверки и сообщения об ошибках во всех частях системы.

Убедитесь, что все соответствующие кнопки / ссылки работают.

Убедитесь, что вы используете соответствующие соглашения об именах для всех частей системы по мере необходимости.

### **5.1 Создание «Мои результаты»**

Создайте форму как указано в «Мои результаты» в презентации.

Эта форма позволяет участнику (который в настоящее время вошел в систему) просматривать оценки компетенции, в которой он / она участвует в текущем соревновании, по каждому модулю и суммарно.

Если участник не участвует в текущем соревновании или не имеет баллов, должны выводиться соответствующие сообщения.

Содержимое, отображаемое в этой форме, включает основную информацию участника и информацию о баллах в компетенции, в которой он участвует.

Основная информация о участнике:

- Имя
- Пол
- IdNumber
- Регион в формате (номер региона – наименование региона)
- Округ

Информация о набранных баллах:

- Чемпионат: год проведения – наименование
- Компетенция: Номер – наименование
- Номер участника (формируется как порядковый номер для каждой компетенции отдельного чемпионата)
- Набранный балл по каждому модулю
- Всего набрано баллов

## **5.2 Создание «О чемпионате»**

Создайте форму как указано в «О чемпионате» в презентации.

## **5.3 Создание «История чемпионата»**

Создайте форму как указано в «История чемпионата» в презентации.

В этой форме для незарегистрированных пользователей будут представлены история и развитие движения WorldSkills. Когда пользователь нажимает на изображение должна выводиться дополнительная информация.

## **5.4 Создание «Компетенции чемпионата»**

Создайте форму как указано в «Компетенции чемпионата» в презентации.

Эта форма показывает незарегистрированным пользователям различные области конкурса Worldskills и подробную информацию о каждой компетенции,

участвующих в каждом из блоков компетенций. Данные взяты из файлов TXT и EXCEL в ресурсах.

В навигации в виде дерева меню разделены на 3 уровня. Содержание каждого уровня:

- Имя первого уровня меню называется «Все компетенции».
- Меню второго уровня отображает блоки компетенций, участвующие в конкурсе Worldskills.
- Меню третьего уровня отображает компетенции, участвующие в каждом блоке.

Использование навигации в виде дерева:

- Реализовано сворачивание и разворачивание меню на каждом уровне.
- При выборе меню первого уровня отобразите справа блоки, участвующие в конкурсе Worldskills.
- При выборе одного поля в меню второго уровня отобразите справа Компетенции, участвующие в этом блоке.
- При выборе одной компетенции в меню третьего уровня отобразите справа детали этой компетенции.

## **5.5 Создание «Предыдущие чемпионаты»**

Создайте форму как указано в «5. Предыдущие чемпионаты» в презентации

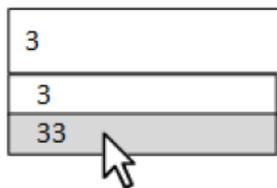
В этой форме информация о предыдущих чемпионатах Worldskills будет показана незарегистрированным пользователям.

В этой форме информация о предыдущих соревнованиях из файла EXCEL будет отображаться в сетке данных, в том числе:

- Порядковый номер
- Год
- Город, регион
- Количество участников

Пользователи могут искать информацию по порядковому номеру, городу или стране. Текстовые поля «Порядковый номер», «Город или страна» должны

поддерживать поиск по нечеткому соответствию с раскрывающимся списком, например: когда в текстовое поле вводится «3», в текстовое поле загружается список с нечетким соответствием, в котором участвует «3», и пользователь может выбрать элемент в этом списке, как показано ниже:



## 5.6 Создание «O WORLDSKILLS RUSSIA»

Создайте форму как указано в «O WorldSkills Russia» в презентации.

В этой форме для незарегистрированных пользователей будут представлена информация об участии WorldSkills Russia в международных чемпионатах. Когда пользователь выберет чемпионат, справа должен отобразиться значок загрузки . Файлы можно загрузить, щелкнув на этот значок (в данном случае должна быть разработана симулирующая загрузка и сохранение файлов на локальном компьютере в указанную пользователем директорию).

## 5.7 Создание «O Краснодарском крае»

Создайте форму как указано в «O Краснодарском крае» в презентации.

В этой форме информация о Краснодарском крае будет показана пользователям с точки зрения истории, событий и туризма, чтобы помочь пользователям узнать о Краснодарском крае.

### **Вкладка «История»:**

История Краснодарского края.

Материалы: History.docx

### **Вкладка «События»:**

Описание крупных событий, прошедших в Краснодарском крае в последние годы. Эти события будут отображаться в порядке убывания в соответствии с временем и продолжительностью их проведения.

Материалы: Event.xlsx

### **Вкладка «Туризм»:**

Информация о живописных местах, отелях и еде Краснодарского края в форме фотографий. При нажатии на пиктограмму должна отображаться большая картинка.

### **5.8 Создание «Управление волонтерами»**

Создайте форму как указано в «Управление волонтерами» в презентации.

Эта форма позволяет координатору управлять волонтерами. Координатор может фильтровать добровольцев в соответствии с их компетенциями, и он может сортировать по всем полям результатов поиска в порядке возрастания. Если поиск не возвращает никаких результатов, должно выводиться соответствующее сообщение.

Общее число волонтеров в списке результатов должно отображаться над сеткой данных.

Содержание в поле со списком «Компетенция» включает в себя: все компетенции, включенные в текущее соревнование (Формат: «Наименование компетенции»).

Информация о добровольцах будет отображаться в списке добровольцев со следующими полями:

- IdNumber
- Имя
- Фамилия
- Пол
- Регион
- Компетенция

Должно выводиться не более 10 записей в таблицу. Для работы со списком должны быть реализованы следующие функции:

- Первая: первая страница
- Назад: предыдущая страница
- Далее: следующая страница
- Последняя: последняя страница

Пользователь так же должен иметь возможность напрямую ввести номер страницы, для перехода на указанную страницу.

В форме должен отображаться номер текущей страницы и общее количество страниц в реальном времени.

Для получения информации о кнопке «Загрузка волонтеров» обратитесь к следующему пункту задания.

### **5.9 Создание «Загрузка волонтеров»**

Создайте форму как указано в «Загрузка волонтеров» в презентации.

Эта форма позволяет координатору импортировать список волонтеров. Координатор может выбрать файл CSV, соответствующий требованиям формата, и нажать «Импорт».

«Импорт» включает в себя следующие проверки и действия:

- Если выбранный файл не является корректным файлом CSV или не соответствует требованиям к формату, должно вывестись соответствующее сообщение.
- Записи импортированных волонтеров должны быть добавлены к существующим данным. Если какая-либо импортируемая запись имеет тот же номер IdNumber с существующей записью, перезапишите существующую запись с этой вновь импортированной записью.
- По завершении импорта должно выводиться общее количество импортированных записей, количество новых записей и количество перезаписанных записей.
- Во время импорта должен показываться индикатор выполнения, чтобы отобразить ход импорта; и система должна реагировать на другие операции пользователя во время импорта.

Обратитесь к материалу структуры и полей импортированного файла CSV.

Материал: volunteers.CSV

### **5.10 Создание «Распределение волонтеров»**

Создайте форму как указано в «18. Распределение волонтеров» в презентации.

Эта форма позволяет координатору корректировать компетенции, на которые распределены волонтеры. Координатор может просматривать информацию о волонтерах, которые закреплены за компетенцией.

Содержимое, отображаемое в таблице, включает:

- Имя
- Пол
- Возраст
- Регион

Координатор может перемещать добровольцев между компетенциями.

Включаются следующие функции:

->: перенести выбранных волонтеров слева направо.

<-: перенести выбранных волонтеров справа налево.

В режиме реального времени отображать под таблицей число волонтеров, задействованных в данной компетенции.

Нажатие кнопки «Сохранить настройку» сохраняет настройку волонтеров в компетенции.

## **Задание 6 Разработка информационной системы учета материалов для производимой предприятием продукции**

В процессе выполнения данного задания вы начнете разработку информационной системы заказчика. Дизайнер предоставил вам набор системной документации, так что вы можете построить систему в соответствии с потребностями клиента.

Создайте базу данных, а затем импортируйте туда необходимые данные.

Затем создайте приложение (часть окон, которые будут доступны пользователям системы) и функциональность для авторизации и оформления заказов.

### **Инструкция участнику**

После выполнения данного задания у Вас должны быть достигнуты результаты, необходимые для того, чтобы заказчик был спокоен, что информационная система будет завершена вовремя.

Убедитесь, что Вы предусмотрели все соответствующие проверки и сообщения об ошибках во всех частях системы.

Убедитесь, что все соответствующие кнопки и механизмы работают. Не забывайте подключать создаваемые Вами формы к формам и интерфейсам, созданным ранее.

Убедитесь, что Вы используете соответствующие соглашения об именах для всех частей системы по мере необходимости.

В конце выполнения задания все результаты будут сохранены и переданы заказчику, без возможности дальнейшей доработки.

### **6.1 Создание базы данных**

Создайте базу данных, используя наиболее подходящую к выданному заданию платформу на сервере баз данных, который Вам предоставлен.

Обратитесь к диаграмме базы данных (ERD) и словарю данных, чтобы создать таблицы согласно спецификации.

Возможно Вам потребуется внесение некоторых изменений для более рационального хранения данных.

После создания базы данных требуется импортировать данные о тканях и фурнитуре. Эти данные не отформатированы для импортирования непосредственно в базу данных, вам необходимо отформатировать данные и загрузить их в таблицы, которые вы только что создали.

### **6.2 Создание приложения**

Создайте приложение, используя наиболее приемлемую для решения задачи платформу.

### **6.3 Авторизация**

Создайте форму для авторизации зарегистрированных пользователей.

В системе будет четыре различных типа пользователей: Заказчики, Менеджеры, Кладовщики и Дирекция.

Пользователи будут входить в систему, используя эту форму. Они будут вводить свой логин и пароль. При попытке входа проверяется на совпадение в базе данных и должно выдаваться сообщение об ошибке в случае неправильного ввода связки логин/пароль.

После успешного входа пользователя в зависимости от их роли происходит перенаправление в следующие разделы:

- Заказчик: "Экран заказчика"
- Менеджер: "Экран менеджера"
- Кладовщик: "Экран кладовщика"
- Дирекция: "Экран директора"

Когда пользователь нажимает на кнопку "Выход" в любом окне / форме системы, пользователи должны быть отключены от системы и совершен переход на Форму авторизации.

Необходимо предусмотреть возможность регистрации заказчиков. Это необходимо для совершения новых заказов.

#### **6.4 Учет материалов**

Сфера деятельности предприятия предполагает специфические требования, предъявляемые к учету материалов:

1. Для каждой номенклатуры может быть задано несколько единиц измерения.
2. Одна из этих единиц измерения должна является учетной, причем для разных материалов учетная единица может быть своей. Например, для тканей это будут рулоны, для пуговиц – килограммы, для украшений – штуки.

3. Пользователи информационной системы (с ролью кладовщика, менеджера и директора) должны иметь возможность получить информацию об остатках, пересчитанных в нужную единицу измерения. Например, если единицей учета для ткани является рулон, то менеджер перед принятием заказа должен иметь возможность узнать, сколько квадратных метров ткани имеется в распоряжении.

4. Пользователь должен иметь возможность указать ту единицу измерения, которая ему удобна. Например, если он пришивает к изделию 10 пуговиц, учетная

единица которых килограмм, то он захочет указать именно 10 пуговиц. Все необходимые расчеты система должна сделать самостоятельно.

5. В системе необходимо вести количественно-суммовой учет материалов. Т.е. в каждый момент времени должно быть известно, сколько и какого материала хранится на складах, и какова закупочная стоимость этих материалов. При списании материалов закупочная стоимость списываемых материалов должна рассчитываться по среднему. Например, на остатках числится 17 погонных метров ткани на закупочную сумму 10 115 рублей. При производстве изделий используется 9 метров ткани. Закупочная сумма используемой ткани в этом случае будет равна:  $10\ 115 * 9 / 17 = 5\ 355$  рублей.

6. В процессе работы появляются обрезки материалов, которые невозможно использовать в производстве. Например, после раскроя остаются куски ткани, площадью 0,3 квадратного метра. Такие обрезки необходимо сразу же списывать с остатков. При этом менеджеры должны владеть информацией о том, какой объем материала был списан на обрезки и какова закупочная стоимость этих обрезков. Пользователи приложения должны иметь возможность для каждого материала настроить условия, при которых остаток единицы хранения этого материала переходит в состояние обрезков.

В разрабатываемом Вами приложении создайте форму для отображения списка тканей, форму списка фурнитуры (формы доступны кладовщику), а также форму списка изделий (форма доступна менеджеру и директору) для возможности ведения такого учета (при необходимости дополните интерфейс вспомогательными формами). Обратите внимание на права пользователя. Заказчики не должны иметь возможность редактировать справочники и учетные данные.

## **6.5 Продукция предприятия**

Помимо учета материалов в приложении должна быть предусмотрена возможность учета выпускаемой продукции. По каждому изделию может быть указана его подробное описание, состав, информация об условиях стирки и чистки.

Для корректного планирования закупок для каждого изделия надо сохранять информацию о плановом объеме материалов, которые необходимы для выпуска этого изделия. Сейчас на предприятии для каждого изделия ведется бумажная спецификация, в которой указано, сколько и какого материала надо использовать для его пошива, поэтому необходимо предусмотреть печать спецификации на изделия на офисном принтере.

Ряд выпускаемой продукции является размерной. То есть при пошиве изделия пользователь должен будет указать еще и его размер. От размера будет зависеть объем материалов, которые планово закладываются в спецификацию на изделие.

С течением времени спецификация на изделие может меняться, например, из-за снятия с производства какой-то ткани, поэтому в системе должна сохраняться история изменения спецификации для последующего просмотра.

Также обязательно предусмотреть хранение информации об остатках готовой продукции (по их количествам и закупочным суммам использованных для их изготовления материалов).

Разработайте форму для просмотра отдельного текстильного изделия с графическим изображением и историей изменения спецификации изделия.

## **6.6 Поступление материалов**

Разработайте документ для отражения факта поступления материалов от поставщиков. В одном документе может быть отражен факт поступления большого количества разных материалов от одного поставщика. В документе пользователь должен иметь возможность указать закупаемые материалы, их количества, закупочную цену и сумму. В документе не должно быть ограничений на дубли товаров, так как поставщик может поставить один и тот же товар по разным закупочным ценам.

Обращайте внимание на работу взаимосвязанных реквизитов. Пользователи ожидают, что при изменении количества сумма будет пересчитана автоматически.

После принятия документа к учёту, остатки материалов документа должны увеличиться на количество и закупочную сумму, указанную в документе.

Документ, принятый к учету не может быть изменен. С данным документом работает кладовщик. Для работы с документом разработайте отдельную форму.

### **6.7 Конструктор изделий**

Для разработки собственных текстильных изделий в системе должен быть реализован «Конструктор изделий».

В «Конструкторе изделий» должны быть реализованы следующие возможности:

- Указание ширины изделия;
- Указание высоты изделия;
- Выбор варианта ткани;
- Выбор варианта окантовки;
- Выбор варианта фурнитуры;
- Выбор размещения и размеров фурнитуры;
- Свободный поворот фурнитуры.

Пользователь должен иметь доступ к двумерному полю, на котором предоставлен схематический предварительный просмотр изделия, включая графическое отображение выбранных материалов.

Пользователь должен иметь возможность с помощью мыши:

- Менять ширину и высоту изделия;
- Менять вариант ткани и окантовки;
- Используя drag&drop размещать и поворачивать фурнитуру на изделии.

При выборе варианта ткани или фурнитуры, пользователь должен иметь возможность добавить свой вариант, путем выбора файла графического изображения.

### **Задание 7 Создание функциональности для учета материалов и готовых изделий, необходимой для заказчика**

В рамках этого задания нужно будет создать функциональность для учета материалов и готовых изделий, необходимую для заказчика.

Разрабатываемая Вами функциональность позволит вести учет материалов, осуществлять закупки материалов, задавать спецификации на производимую предприятием продукцию, отражать в системе факт выпуска продукции, проводить инвентаризацию и получать управленческую информацию в виде отчетов.

### **Инструкция участнику**

К концу выполнения задания у Вас должны быть достигнуты результаты, необходимые для того, чтобы заказчик был спокоен, что информационная система будет завершена вовремя.

Убедитесь, что Вы предусмотрели все соответствующие проверки и сообщения об ошибках во всех частях системы.

Убедитесь, что все соответствующие кнопки и механизмы работают. Не забывайте подключать создаваемые Вами формы к формам и интерфейсам, созданным ранее.

Убедитесь, что Вы используете соответствующие соглашения об именах для всех частей системы по мере необходимости.

В конце разработки все результаты будут сохранены и переданы заказчику, без возможности дальнейшей доработки.

## **7.1 Инвентаризация**

Из-за всевозможных ошибок в учете регулярно возникает задача инвентаризации, то есть пересчета реальных остатков материалов и уже готовой продукции и приведение учетных данных в соответствие с реальностью. Проведение инвентаризации и надо автоматизировать.

Планируется следующая схема работы. При проведении инвентаризации сотрудники предприятия идут на склад и пересчитывают абсолютно все находящиеся на складе материалы и изделия. Далее собранные сотрудниками данные вводятся в приложение. В форме показываются расхождения между введенными данными и данными учетной системы. Если расхождения между реальными и учетными данными по закупочным суммам превышают 20%, то это говорит о том, что

инвентаризация могла быть проведена не качественно и посчитан не весь товар. Принять такой документ к учету можно, только после утверждения директором.

Все выявленные в процессе инвентаризации излишки должны быть оприходованы, все выявленные в процессе инвентаризации недостачи должны быть списаны. Закупочные суммы списания должны считаться по среднему.

## **7.2 Стандартные управленческие отчеты**

Для удобства принятия решения, руководству и персоналу предприятия надо иметь возможность оперативно получать следующую информацию:

1. Информацию об остатках материалов / изделий на предприятии. Остатки могут быть получены как по всем материалам / изделиям, так и по выбранным пользователем.

2. Информацию о движении материалов / изделий за период. То есть сколько материалов / изделий было на начало выбранного пользователем периода, сколько было поступлений, списаний и какой получился конечный остаток материала / изделия.

Разработайте формы для отображения стандартных управленческих отчетов с возможностью вывода отчетов на печать. Данные формы должны быть доступны директору.

## **7.3 Список заказов**

Вам нужно разработать форму отображения списка заказов. На этой форме должны отображаться:

- Номер и дата заказа;
- Суммарное количество изделий заказа;
- Этап выполнения;
- Заказчик;
- Менеджер.

Каждый заказ может проходить несколько этапов обработки менеджерами (или иными ответственными сотрудниками).

Каждому этапу обработки заказа соответствует его «статус» - краткое однозначное отображение пользователю информации, в каком состоянии находится заказ. Количество статусов со временем развития информационной системы может меняться.

Сейчас каждый заказ может проходить следующие этапы обработки:

1. Только что созданный заказчиком заказ. Такому заказу автоматически присваивается статус «Новый». Заказчик может изменять данные в заказах только со статусом «Новый».

2. Отправленный с мобильного устройства в центральную базу заказ, еще не принятый к обработке менеджером получает статус «Ожидает».

3. Когда менеджер приступает к проверке заказа, ему присваивается статус «Обработка».

4. Если менеджер отклоняет заказ, ему назначается статус «Отклонен» и к дальнейшей обработке он становится непригоден.

5. После того как менеджер одобрит заказ, ему присваивается статус «К оплате».

6. После поступления оплаты, заказу присваивается статус «Оплачен» и он передается на раскрой (выполнение).

7. Когда система включила его в раскрой и передала его на выполнение, заказу назначается статус «Раскрой».

8. Заказу, полностью обработанному системой, назначается статус «Готов».

Последовательность статусов не может быть изменена.

#### **7.4 Оформление заказов**

Разработайте форму, позволяющую оформлять заказ на производство текстильных изделий. Данный функционал доступен только для Заказчика и Менеджера.

Форма должна позволять выбирать изделия, которые необходимо создать при выполнении заказа и количество этих изделий.

Важно, что в одном заказе может быть несколько разных изделий. Пользователь должен иметь возможность конструировать изделия самостоятельно, воспользовавшись конструктором изделий.

Для каждой позиции заказа рассчитывается ее стоимость как количество \* цену (цена складывается из стоимости всех материалов, затрачиваемых на изготовление изделия).

Для всего заказа должна быть рассчитана итоговая стоимость – суммарная стоимость всех позиций заказа.

Так как заказчик или менеджер могут быть плохо знакомы с компьютером, форма для ввода данных о заказе должна быть как можно более простой и максимально исключать ошибки ввода данных.

Важно, что заказчик может изменять заказ до проверки менеджером.

Когда заказчик самостоятельно оформляет заказ, то этот заказ отображается у всех менеджеров системы. Как только один из менеджеров открывает заказ, то этот менеджер записывается как менеджер заказа. Далее данный заказ отображается только у заказчика и ответственного менеджера.

## **7.5 Производство изделий**

Разработайте форму, позволяющую отражать факт производства продукции. Форма должна отражать факт производства некоторого количества какого-то одного изделия конкретного размера. На форме необходимо отражать информацию об объеме фактически использованных для пошива материалов. На форме обязательно должен отражаться факт превышения объема фактически использованных материалов от планового более чем на 15%, так как это говорит об ошибке в плановой спецификации или ошибке раскроя.

Себестоимость произведенных изделий считается как общая закупочная сумма всех материалов, которые были использованы при производстве этих изделий.

Так как персонал, занятый на производстве изделий, может быть плохо знаком с компьютером, форма для ввода данных о реально использованных материалах при пошиве изделия должна быть как можно более простой и максимально исключать ошибки ввода данных.

В процессе производства изделий неизбежно возникают обрезки материалов. При сохранении данных формы, информация об образовавшихся обрезках должна быть рассчитана и принята к учету автоматически.

### **Задание 8 Создание части системы, пригодной для автоматического раскроя изделий отдельного заказа**

В рамках выполнения этого задания нужно будет создать часть системы, пригодную для автоматического раскроя изделий отдельного заказа.

Разрабатываемая Вами функциональность позволит минимизировать затраты производственного персонала на раскрой изделий и оптимизировать расход ткани путем автоматического распределения заказанных изделий на ткани.

Также в рамках сессии Вам предстоит разработать проект развития информационной системы.

#### **Инструкция участнику**

К концу выполнения задания у Вас должны быть достигнуты результаты, необходимые для того, чтобы заказчик был спокоен, что информационная система будет завершена вовремя.

Убедитесь, что Вы предусмотрели все соответствующие проверки и сообщения об ошибках во всех частях системы.

Убедитесь, что все соответствующие кнопки и механизмы работают. Не забывайте подключать создаваемые Вами формы к формам и интерфейсам, созданным ранее.

Убедитесь, что Вы используете соответствующие соглашения об именах для всех частей системы по мере необходимости.

В конце все результаты будут сохранены и переданы заказчику, без возможности дальнейшей доработки.

## **8.1 Проектирование развития системы**

Вам поручили разработать дизайн базы данных, а также диаграмму прецедентов для дальнейшего развития информационной системы. Вам не нужно разрабатывать систему, Вам необходимо только создать диаграмму «сущность-связь» и диаграмму прецедентов.

## **8.2 Оценка затрат ткани**

Одной из основных задач предприятия является снижение издержек производства. Одним из способов является сокращение потерь материалов, возникающих в процессе раскроя тканей.

Ткани поступают на предприятие в виде рулонов определенной ширины и длины. Для того чтобы использовать эти исходные материалы, их требуется раскроить на части нужных размеров и формы. При этом, как правило, некоторая часть материалов идет в отходы. Эти отходы часто не используются, либо используются не полноценно. В связи с этим необходимо разработать модуль раскроя тканей, позволяющий минимизировать отходы.

**Важно, что все изделия могут быть только прямоугольной формы.**

**Размещение изделий под наклоном недопустимо.**

Предлагается следующий алгоритм сокращения издержек производства:

1. При поступлении заказа все необходимые изделия группируются по тканям, из которых они изготавливаются.
2. Изделию нумеруются в своей группе в порядке убывания площади.
3. Внутри каждой группы, изделия начиная от самого большого к самому маленькому (по площади) размещаются на ткани либо горизонтально, либо вертикально, учитывая, что расход рулона должен быть минимальным.
  - a. Выбирается первое наибольшее по площади изделие.
  - b. Со склада выбираются все остатки и рулоны.
  - c. Если ширина остатка (предпочтительно) или рулона позволяет, то изделие размещается горизонтально, иначе – вертикально.

d. Выбирается следующее изделие и предыдущий шаг повторяется.

e. Если для изготовления изделий недостаточно ткани, то указывается то количество, которое возможно изготовить.

f. По итогам раскроя необходимо вывести информацию о том сколько изделий из заказа можно выполнить и затраты ткани для каждой позиции заказа.

Разработайте форму, позволяющую производить автоматический раскрой по приведенному алгоритму. Данная форма должна быть доступна только менеджеру.

Вы вправе предложить собственный алгоритм раскроя.

### **8.3 Тестирование**

Для выполнения процедуры тестирования функциональности автоматического раскроя Вам нужно описать пять вариантов тестирования.

Каждый из этих вариантов тестирования должен добавлять в систему новый заказ, который подлежит раскрою. В заказе может быть несколько позиций. Каждое изделие заказа может содержать или не содержать фурнитуру. Заказы могут быть выполнимы или невыполнимы по причине отсутствия необходимого количества ткани или фурнитуры. Желательно, чтобы варианты тестирования демонстрировали различные исходы работы алгоритма.

### **Задание 9 Создание информационной системы заказчика, опираясь на задания 6, 7, 8**

В рамках этого задания нужно будет создать функциональность для печати бумажных форм документов, необходимую для заказчика.

Разрабатываемая Вами функциональность позволит выполнять печать документов для ведения бумажной копии учета.

#### **Инструкция участнику**

К концу выполнения задания у Вас должны быть достигнуты результаты, необходимые для того, чтобы заказчик был спокоен, что информационная система будет завершена вовремя.

Убедитесь, что Вы предусмотрели все соответствующие проверки и сообщения об ошибках во всех частях системы.

Убедитесь, что все соответствующие кнопки и механизмы работают. Не забывайте подключать создаваемые Вами формы к формам и интерфейсам, созданным ранее.

Убедитесь, что Вы используете соответствующие соглашения об именах для всех частей системы по мере необходимости.

В конце работы все результаты будут сохранены и переданы заказчику, без возможности дальнейшей доработки.

### **9.1 Построение карты раскроя**

Для передачи заказа на производство требуется разработать форму, содержащую графическое изображение раскроя изделий в масштабе.

Форма должна отображать все требуемые остатки и рулоны тканей с нанесенным контуром изделий, содержащихся в заказе.

Нужно предусмотреть возможность вывода изображения на печать.

Заказчик должен иметь возможность следить за изменением состояния заказа.

### **9.2 Оценка затрат фурнитуры**

Изделие может содержать различную фурнитуру. Так как от наличия фурнитуры зависит возможность изготовления изделия, то необходимо реализовать функционал по проверке всей требуемой для выполнения заказа фурнитуры.

Создайте форму, отображающую информацию о наличии фурнитуры для каждой позиции заказа. Если некоторой фурнитуры недостаточно для изготовления текстильных изделий, то необходимо вывести информацию о том, сколько изделий не может быть выполнено и какой фурнитуры недостаточно. Данная форма должна быть доступна только менеджеру.

### **9.3 Отчет по списанию материалов/изделий**

Руководство предприятия хочет видеть информацию об объеме материалов, которые были списаны не продуктивно: на обрезки, вследствие недостат при проведении инвентаризации, из-за ошибок раскроя. Необходимо видеть, как количество списанных материалов, так и их стоимость.

Заранее неизвестно как пользователям будет удобно анализировать данные: видеть по каждой причине списания какие товары были списаны, или по каждому товару видеть какие по нему были выявлены причины списания и в каком объеме, в графическом виде это будет удобно анализировать или в текстовом.

Необходимо предусмотреть наиболее универсальные возможности для отображения такой информации.

#### **9.4 Отчет по остаткам материалов в разрезе единиц**

Пользователям хочется видеть информацию об остатках материалов в разрезе их единиц измерения. Например, для тканей хочется оперативно понимать, сколько рулонов ткани есть на остатках и сколько это квадратных метров. Необходимо предусмотреть в приложении отчет, который позволит видеть остатки материалов их единицах измерения.

#### **Задание 10 Разработка приложения и базы данных для Marathon Skills**

Marathon Skills проводится каждый год в разных частях мира. Он открыт для всех бегунов различных степеней опыта, и часто включает в себя следующие виды зачетов: 5 км- для новичков, 21км - полумарафон и полный марафон - 42км.

Эта система будет работать как общественная система (например, позволяя людям, зарегистрироваться на марафон, выбрать спонсора, узнать больше о марафоне, посмотреть результаты марафона) и закрытой части (например, позволяя персоналу управлять регистрацией спортсменов, учитывать волонтеров, учитывать спонсорство).

##### **Пользователи**

Система предназначена для различных типов пользователей:

Внешние пользователи (не зарегистрировались в системе);

Бегуны;

Координаторы;

Администраторы.

### **Месторасположение системы**

Есть два основных места где будет использоваться система:

1. Штаб-квартира

Marathon

Skills

Координаторы и администраторы будут работать в системе через офисные компьютеры.

2. Мобильный компьютерный киоск

Чтобы дать возможность заинтересовавшимся людям легко принять участие, Организаторы ездят по разным местам, таким как спортивные клубы, университеты и торговые центры с мобильным киоском (см фото ниже). Это позволит заинтересованным людям, узнать больше о марафоне, а также зарегистрироваться в качестве бегуна или спонсировать бегуна.

В рамках данного задания вы начнете разработку приложения и базы данных для Marathon Skills 2016. Дизайнер предоставил вам набор системной документации, так что вы можете построить систему в соответствии с потребностями клиента. Найдите время для знакомства с предоставленными материалами.

Создайте базу данных, а затем импортируйте туда необходимые данные.

Затем создайте приложение: часть окон, которые будут доступны пользователям системы.

### **Инструкция участнику**

К концу выполнения задания, у вас должны быть достигнуты следующие результаты, необходимые для того, чтобы заказчик был спокоен, что система будет завершена вовремя.

Убедитесь, что вы следуете предоставленному руководству по стилю во всех частях системы.

Убедитесь, что вы предоставляете соответствующие проверки и сообщения об ошибках во всех частях системы.

Убедитесь, что все соответствующие кнопки / ссылки работают.

Убедитесь, что вы используете соответствующие соглашения об именах для всех частей системы по мере необходимости.

### **10.1 Создание базы данных**

Создайте базу данных, используя знакомую вам платформу на сервере баз данных, который вам предоставлен.

### **10.2 Загрузка данных**

Сценарий SQL предоставлен для вас, чтобы создать большинство таблиц и вставки данных в них. Все, что вам нужно сделать, это импортировать сценарий SQL в вашу базу данных.

### **10.3 Создать таблицы для персонала согласно спецификации**

Обратитесь к диаграмме базы данных (ERD) и словарю данных.

Создайте таблицы сотрудников (персонал, положение и расписаний) согласно спецификации.

### **10.4 Импорт данных персонала**

Все данные сотрудников были представлены в `marathon-skills-2016-staff-import`

Эти данные не отформатированы для импортирования непосредственно в базу данных, вам необходимо отформатировать данные и загрузить их в таблицы, которые вы только что создали.

Поле Summary Information не требуется.

В поле " Full Name" в формате "Имя Фамилия" используются разные символы разделителя.

Убедитесь, что адреса электронной почты в правильном формате.

### **10.5 Создать приложение**

Создайте приложение, используя выбранную вами платформу

## **10.6 Создание "Главный экран системы"**

Создайте главное меню системы "Главный экран системы".

Каждое окно / страница приложения, который имеет "? дней? часов и? минут до начала гонки в "в нижней части экрана должно автоматически обновляться в режиме реального времени. Рассчитать количество времени, оставшегося до начала первого Marathon Skills начинается (2017-11-24 6:00).

## **10.7 Создание окна «Подробная информация»**

Создание окна "Подробная информация".

## **10.8 Создать окно "Меню авторизации"**

Создать регистрационную форму "Меню авторизации".

Там будет три различных типа пользователей данной системы - Бегуны, Координаторы и Администратор

Люди будут входить в систему, используя эту форму. Они будут вводить свой адрес электронной почты и пароль (хранится в базе данных в незашифрованном виде), при попытке входа проверяется на совпадение в базе данных и должно выдаваться сообщение об ошибке в случае неправильного ввода связки логин/пароль.

После успешного входа пользователя в зависимости от их роли происходит перенаправление в следующие разделы:

- Бегун: "Меню Бегуна"
- Координатор: "Меню Координатора"
- Администратор: "Меню Администратора"

Когда пользователь нажимает на кнопку "Выход" в любом окне / форме системы, пользователи должны быть отключены от системы и совершен переход на окно "Главный экран системы".

## **10.9 Создать окно "Меню бегуна"**

Создать меню "Меню Бегуна".

Если пользователь входит в систему как бегун он должен быть перенаправлен в это меню.

Пожалуйста, обратите внимание - вы будете строить эту систему постепенно, и некоторые функции могут быть добавлены в следующих заданиях.

### **10.10 Создать окно "Меню Координатора"**

Создать меню "Меню координатора".

Если пользователь входит в систему как координатор он должен быть перенаправлен в это меню.

Пожалуйста, обратите внимание - вы будете строить эту систему постепенно, и некоторые функции могут быть добавлены в следующих заданиях.

### **10.11 Создать "Меню администратора "**

Создать меню "Меню администратора".

Если пользователь входит в систему как администратор он должен быть перенаправлен в это меню.

Пожалуйста, обратите внимание - вы будете строить эту систему постепенно, и некоторые функции могут быть добавлены в следующих заданиях.

### **10.12 Создать окно "Проверки уже введенных раньше бегунов "**

Создать окно "Проверки уже введенных раньше бегунов".

Это меню дает бегунам выбор, чтобы зарегистрироваться в качестве нового бегуна, или указать, что они регистрировались как бегун ранее.

### **10.13 Создать форму «Информация о Marathon Skills»**

Создать форму «Информация о Marathon Skills».

Эта страница содержит информацию о Marathon Skills, отображает фотографии и предоставляет пользователям доступ к интерактивной карте.

**Задание 11 Разработка функций авторизации в системе для бегунов на базе задания 10**

Создать функциональность авторизации в системе, что позволит бегунам зарегистрироваться в «Marathon Skills», и позволит бегунам редактировать свои профили, как только они вошли в систему.

### **Инструкция участнику**

К концу выполнения задания, у вас должны быть достигнуты следующие результаты, необходимые для того, чтобы заказчик был спокоен, что система будет завершена вовремя.

Убедитесь, что вы следуете предоставленному руководству по стилю во всех частях системы.

Убедитесь, что вы предоставляете соответствующие проверки и сообщения об ошибках во всех частях системы.

Убедитесь, что все соответствующие кнопки / ссылки работают.

Убедитесь, что вы используете соответствующие соглашения об именах для всех частей системы по мере необходимости.

## **11.1 Создать окно "Меню регистрации бегунов "**

Создайте форму «Меню регистрации бегунов».

Измените базу данных таким образом, чтобы в ней была возможность хранить фотографию бегуна

Когда бегун вносит всю информацию, должна быть создана учетная запись пользователя с ролью "Бегун". Эти пользователи будут иметь возможность входа в систему.

- Все поля обязательны для заполнения.
- Выбор пола и список стран, должны браться из базы данных.
- E-mail адрес должен быть в правильном формате, например, x@x.x
- Пароль должен отвечать следующим требованиям:
  - Минимум 6 символов
  - Минимум 1 прописная буква
  - Минимум 1 цифра
  - По крайней мере один из следующих символов: ! @ # \$ % ^

- Значение "повторите пароль" должно соответствовать с значением "Пароль".

"Дата рождения" должна быть правильной датой и бегуну должно быть не менее 10 лет на момент регистрации.

Фотография бегуна должна отображаться в форме. Должна быть возможность загрузить фотографию бегуна из любого места на компьютере. Должна быть возможность изменить фотографию, на другую.

## **11.2 Создать форму "Регистрация на марафон "**

Создайте форму «Регистрация на марафон».

Эта форма позволяет бегуну, который регистрируется на марафон (или несколько марафонов) в Marathon Skills 2016, выбрать виды марафонов, в которых он хочет участвовать, выбирать гоночный комплект (RFID браслет + нагрудник бегуна + дополнительные оплачиваемые предметы) и выбрать благотворительные организации, которые он хочет поддержать. Общая стоимость должна рассчитываться.

- По крайней мере 1 вид марафона должен быть выбран.
- "сумма взноса" должна быть действительным положительным числом.
- Нажатие на кнопку информации (i) рядом с взносом будет отображать информацию благотворительной организации во всплывающем окне.
- "Стоимость регистрации" рассчитывается на основе стоимости любых выбранных событий + стоимость опциона гоночного комплекта

## **11.3 Создать форму "Подтверждение регистрации бегуна "**

Создать окно "Подтверждение регистрации бегуна".

Эта страница будет показана бегуну, как только он успешно зарегистрировался на какое-либо событие.

## **11.4 Создать окно "Редактирование профиля"**

Создать окно "Редактирование профиля".

Эта форма позволяет бегуну редактировать свой профиль. Он может изменить свои личные данные и изменить свой пароль. Бегун не может изменить свой адрес

электронной почты, поскольку он используется в качестве имени пользователя для входа в систему.

Описание.

- Все поля обязательны для заполнения.
- Выбор пола и список стран, должны браться из базы данных.
- Пароль должен отвечать следующим требованиям:
  - Минимум 6 символов
  - Минимум 1 прописная буква
  - Минимум 1 цифра
  - По крайней мере один из следующих символов: ! @ # \$ % ^
- Значение "повторите пароль" должно соответствовать значению "Пароль".

"Дата рождения" должна быть правильной датой и бегуну должно быть не менее 10 лет на момент регистрации.

Фотография бегуна должна отображаться в форме. должна быть возможность загрузить фотографию бегуна из любого места на компьютере. Должна быть возможность изменить фотографию, на другую.

### **11.5 Создать форму "Мои результаты участия в марафонах"**

Создать форму "Мои результаты участия в марафонах".

Это окно показывает бегуну (который авторизовался в системе) его результаты с предыдущих соревнований. Если они не соревновались раньше, должно выводиться соответствующее сообщение.

Должен отображаться пол, возраст и категория бегуна. Бегун впишется в одну из следующих возрастных категорий:

- до 18
- от 18 до 29
- от 30 до 39
- от 40 до 55
- от 56 до 70
- более 70

Список должен показать каждое соревнование, в котором бегун соревновался ранее.

Выводятся следующие поля для каждого события:

- Марафон: полное название марафона.
- Событие: полное название события.
- Время: время гонки бегуна на события в часах, минутах и секундах.
- В целом положение бегуна в марафоне среди всех участников.
- Отдельно по категории положение бегуна на событии, среди бегунов одного того же пола и той же возрастной категории.

Чем лучше время, тем ниже ранг. Например, лучшему времени марафона присваивается первое место

- Если несколько бегунов закончили с одинаковым временем, им должно быть присвоено одинаковое место. Например:

<b>ЗВАНИЕ</b>	<b>ГОНКА ВРЕМЯ</b>
#1	4ч 15м 12с
#2	4ч 22м 23с
#2	4ч 22м 23с
#4	4ч 25м 45с

При нажатии кнопки "показать все результаты" должен осуществляться переход к "Результаты прошедших гонок".

## **11.6 Создать форму "Насколько долгий марафон "**

Создать форму "Насколько долгий марафон".

Это интерактивный раздел приложения, показывающий зрителям, как долго длится 42km полный марафон. Пользователь выбирает или «скорость», или "расстояние", а затем выбирает элемент в этой категории. Детали этого элемента будут отображаться, а также показываться расчетное значение на основании скорости или длины выбранного элемента.

Если пункт "Скорость" выбран:

- Отображение имени элемента и фото.

- Дисплей: "Максимальная скорость AAA BBB. Это займет CCC чтобы завершить 42km марафон".

- AAA: название пункта.

- BBB: максимальная скорость выбранного элемента в км / ч.

- CCC: время, необходимое для того чтобы пункт совершил путешествие на расстояние 42km на максимальной скорости в несколько часов или минут.

Если пункт "Расстояние" выбран:

- Отображение имени элемента и фото.

- Дисплей: "Длина AAA BBB. Это займет CCC из них, чтобы покрыть расстояние в 42 км марафона".

- AAA: название пункта.

- BBB: длина элемента в метрах.

- CCC: количество элементов, которые будут необходимы, чтобы покрыть расстояние в 42км.

## **11.7 Создать форму "Интерактивная карта"**

Создать форму "Интерактивная карта".

Эта форма является интерактивной особенностью приложения, предоставляет информацию о ходе Marathon Skills для зрителей. Пользователь может просматривать карту и нажать на различных выделенные точки карты (контрольно-пропускные пункты и события, начальные позиции) отмеченные на трассе. При нажатии, показываются детали этой точки. Могут отображаться, например, ближайшие ориентиры и любые услуги, которые на точке предоставляются (напитки, энергетические батончики, туалеты, информационные и медицинские пункты).

Убедитесь, что восемь контрольно-пропускных пунктов равномерно расположены вокруг марафона (их точное положение не слишком важно).

Убедитесь, что все 3 варианта дистанций, отмечены (предположим, что марафон протяженностью 42km и, что бегуны будут работать в направлении по часовой стрелке):

1. Начало 42 км полный марафон - в начале / окончании курса (отмечены клетчатым флагом).

2. Начало 21km Half Marathon - на полпути курса.

3. Начало 5км Fun Run - 5 км до конца курса.

### **11.8 Создайте форму "Управление информацией о бегунах"**

Создать окно "Управление информацией о бегунах".

Когда координатор авторизовался в системе, он должен иметь возможность просматривать список всех бегунов, которые зарегистрированы на текущий марафон. Координатор может фильтровать бегунов по статусу регистрации и выбранным типам марафона, и он может сортировать все поля таблицы результатов. Если их поиск не дал никаких результатов, должно выводиться соответствующее сообщение.

Общее количество бегунов, которые выведены в список, должно быть отображено над списком.

Имя, Фамилия, адрес электронной почты и регистрационный статус должны быть указаны для каждого бегуна.

При нажатии кнопки "Редактировать" рядом с бегуном загружается окно "23. Управление бегунами" в которой выводится более подробная информация о выбранном бегуне.

См пункт 11.9 для информации о кнопке "Детальной информация (CSV)".

Кнопка «E-mail список» См пункт 11.10 для информации о создании выгрузки адресов e-mail.

### **11.9 Создание выгрузки дополнительной информации о бегунах**

Когда пользователь нажимает кнопку "Детальная информация (CSV)" в форме "Управление информацией о бегунах", выгружается CSV файл данных (значения, разделенные запятыми), содержащий всю необходимую информацию обо всех бегунах, которые в настоящее время отображены в списке.

У пользователя должен быть выбор места выгрузки для файла CSV.

По каждому бегуну выводится следующая информация:

- Имя
- Фамилия
- Эл. адрес
- Пол
- Страна
- Дата рождения
- Возраст (полный) в годах на день проведения марафона
- Состояние регистрации
- выбранный комплект
- Тип марафонов (объединенные в одну строку, разделенные запятыми, например, 5 км для новичков, 21км полумарафон).

Должна выводиться информация в файл только по тем бегунам, которые были отобраны пользователями в список.

### **11.10 Создание выгрузки информации о электронной почте бегунов**

Когда пользователь нажимает кнопку "E-mail список" в форме "16. Управление информацией о бегунах ", выгружается список адресов электронной почты бегунов, которые они могут скопировать и вставить в клиент электронной почты.

Список адресов электронной почты должен показываться во всплывающем окне, так чтобы содержимое можно было скопировать и вставить в другую программу.

Список адрес электронной почты должен использовать следующий формат для отображения фамилии, имени и адреса электронной почты:

"Фамилия Имя 1" <email@address.com>; "Фамилия Имя 2" <email@address.com>;

Список адрес электронной почты должен содержать только информацию о бегунах, которые соответствуют критериям сортировки и фильтрации, указанным пользователем.

## **Задание 12 Разработка функций авторизации в системе для бегунов на базе задания 11**

Создать функциональность авторизации в системе, что позволит бегунам зарегистрироваться в «Marathon Skills», и позволит бегунам редактировать свои профили, как только они вошли в систему.

### **Инструкция участнику**

К концу выполнения задания, у вас должны быть достигнуты следующие результаты, необходимые для того, чтобы заказчик был спокоен, что система будет завершена вовремя.

Убедитесь, что вы следуете предоставленному руководству по стилю во всех частях системы.

Убедитесь, что вы предоставляете соответствующие проверки и сообщения об ошибках во всех частях системы.

Убедитесь, что все соответствующие кнопки / ссылки работают.

Убедитесь, что вы используете соответствующие соглашения об именах для всех частей системы по мере необходимости.

### **12.1 Встреча с заказчиком**

В ходе этой сессии, вы будете вызваны для участия в короткой (5 минут) встрече с клиентом. Вы должны показать себя с профессиональной стороны на протяжении всей встречи. Клиент будет иметь копию вашей работы с предыдущих сессий.

Вам будет выдана «Повестка встречи»

Данная повестка описывает, что будет интересоваться заказчика, во время вашей встречи.

[marathon-skills-2016-meeting-agenda.pdf](#)

Контрольный список для совещаний по демонстрации системы

Следуйте контрольному списку, чтобы показать себя профессионалом

### **12.2 Создать форму "ВМІ калькулятор"**

Создать форму "ВМІ калькулятор".

Эта форма позволяет пользователю определить их ВМІ индекс на основе их веса и роста.

ВМІ обозначает индекс массы тела. Он используется, чтобы дать вам представление о том, является ли Ваш вес- избыточным или идеальным для вашего роста. Это полезно знать, потому что если ваш вес увеличивается или уменьшается за пределами идеального диапазона, у вас повышается риск заболеть.

Расчетный индекс ВМІ будет отображаться на шкале, как показано в презентации, положение стрелки над шкалой должно показать, в какой категории пользователь находится в (недостаточный вес, здоровый вес, избыточный вес или ожирение).

Значок человека над шкалой должен изменяться в зависимости от категории, к которой относится пользователь (недостаточный вес, здоровый вес, избыточный вес или ожирение) и соответствующая надпись в значке должна отображаться.

Пользователь выбирает свой пол, нажав на мужскую или женскую иконку. Выбранный значок должен иметь более толстую границу или должен другим способом показывать, что он выбран.

Примечание: пол запрашивается в форме, но не используется в расчетах.

Требуется ввод в следующие поля:

- Вес (кг)
- Рост (см)

Расчет:

$$\text{ВМІ} = \text{вес} / (\text{рост} \times \text{рост})$$

(вес в кг, рост в м)

Вывод индекса ВМІ в категориях:

<18,5 недостаточный вес

18.5 - 24.9 здоровый вес

25 - 29,9 Избыточный вес

> 30 Ожирение

### **12.3 Создать "17. Управление бегуном "**

Создать форму "Управление бегуном ".

В этом окне отображается дополнительная информация о выбранном в форме «Управление информацией о бегунах» бегуне.

Все личные данные бегуна должны отображаться, а также информация, по регистрационному статусу на текущий марафон.

Состояние регистрации бегуна должно быть показано ясным визуальным способом с использованием значков. Значок галочка с текущего статуса бегуна и на всех значках выше. Значок крест должен быть показан ниже текущего статуса и по всем полям ниже.

Правильный порядок статусов:

1. Зарегистрирован
2. Оплата подтверждена
3. Выдан комплект
4. Вышел на старт

Должна отображаться фотография бегуна

Когда нажата кнопка "Показ сертификата", то должна открываться форма "19. Показ сертификата" с правильными данными.

Когда нажата кнопка "Редактирование профиля" то осуществляется переход к форме "18. Редактирование профиля бегуна" для того, чтобы координатор смог отредактировать информацию о бегуне.

Когда нажата кнопка «Печать бейджа», то должна открыться печатная форма бейджа, в бейдж должна выводиться следующая информация: Фотография бегуна, Фамилия Имя бегуна, флаг страны бегуна, название страны бегуна, В каких типах забега принимает участие. А также название благотворительной организации, за которую он бежит.

### **12.4 Создать форму "Редактирование профиля бегуна"**

Создать форму "Редактирование профиля бегуна ".

Эта форма позволяет координатору редактировать профиль бегуна. Он может изменить личные данные бегуна, изменить его статус и регистрацию на марафон и изменить пароль. Адрес электронной почты не может быть изменен, так как он используется в качестве имени пользователя для входа в систему.

- Все поля обязательны для заполнения.
- Список полов и список стран, выбираются из базы данных.

Пароль должен отвечать следующим требованиям:

- Минимум 6 символов
- Минимум 1 прописная буква
- Минимум 1 цифра
- По крайней мере один из следующих символов: ! @ # \$ % ^
- Значение "повторите пароль" должно соответствовать значению "Пароль".

"Дата рождения" должна быть правильной датой и бегуну должен быть не менее 10 лет на момент регистрации.

- Фотография бегуна должна отображаться в форме. должна быть возможность загрузить фотографию бегуна из любого места на компьютере. Должна быть возможность изменить фотографию, на другую.

## **12.5 Создать форму "Показ сертификата"**

Создать форму "Показ сертификата".

Это окно позволяет координатору открывать для предварительного просмотра любой сертификат, который получил бегун, в предыдущем марафоне, например Marathon Skills 2014.

Координатор выбирает дистанцию в которой бежал бегун (в окне выбора должен появляться список только тех дистанций, на которые бегун был зарегистрирован) и предварительный просмотр сертификата, который был выдан бегуну и он отображается. Если бегун не имеет права на получение сертификата (например, он не финишировал или не был зарегистрирован на марафон), должно выводиться соответствующее сообщение.

Логотип Marathon Skills 2014 должен быть на видном месте: marathon-skills-2014-logo.png

На сертификате также должен быть представлен и символ: marathon-skills-2014-certificate-seal.png

Вся выводимая информация должна быть рассчитана и показана как в презентации, в том числе:

- Имя бегуна
- Название зачета
- Время гонки в часах, минутах и секундах
- Общее место участника в зачете
- Наименование марафона и место проведения
- Какое количества спонсорских денег привлек бегун

Наименование благотворительной организации.

## **12.6 Тестирование**

Вас попросили заполнить несколько тестов части приложения, которые были разработаны для этой системы.

Используйте прилагаемый шаблон тестирования, чтобы определить 10 тестов, которые проверяют любые из форм в вашем приложении, например: Войти, спонсорство, регистрация бегуна, добавление / редактирование пользователей. Убедитесь, что вы заполнили все соответствующие части шаблона тестирования для каждого теста. Ваша документация должна явно показывать, какую часть приложения, вы тестируете.

**Формат Демонстрационного экзамена в соответствии со Спецификацией стандарта компетенции № 09 «Программные решения для бизнеса»**

Формат Демонстрационного экзамена: **Очный.**

Форма участия: **Индивидуальная.**

Вид аттестации: **ГИА, Промежуточная.**

## **Список оборудования и материалов, запрещенных на Демонстрационном экзамене:**

- Запрещено использование клавиатур и мышек с подключением по беспроводным каналам. Устройства ввода не должны быть программируемыми.
- Участникам запрещено приносить:
  - Дополнительные программы,
  - Мобильные телефоны,
  - Портативные электронные устройства (планшеты, и т. п.),
  - Смарт-часы,
  - Устройства для хранения информации (флэш-накопители, диски, и т.п.).
- Оборудование не должно иметь доступ к внутренним устройствам для хранения информации.
- Эксперты обладают правом запретить определенное оборудование в зоне проведения экзамена.
- Экспертам запрещено без согласования с Главным экспертом пользоваться личными компьютерами, планшетами или мобильными телефонами, находясь в помещении для экспертов, когда относящиеся к экзамену документы находятся в комнате.
- Участникам и экспертам запрещается использовать личные устройства для фото- и видеосъемки на площадке проведения до завершения экзамена без согласования с Главным экспертом.

### **4.1 Типовые практические задания:**

Все практические задания выполняются на предложенной базе данных

#### **4.1.1 Запросы к одиночным таблицам**

Задача 1. Напишите запрос к таблице Sales.Orders, который возвращает заказы, размещенные в последний день месяца.

Задача 2. Напишите запрос к таблице Sales.OrderDetails, который возвращает заказы с общей стоимостью (вычисляется как количество, умноженное на цену отдельного товара) более 10 000. Сортировка выполняется по общей стоимости.

#### **4.1.2 Соединения**

**Задача 3.** Запрос должен вернуть всех клиентов, которые не делали заказов. Используются таблицы Sales.Customers и Sales.Orders.

**Задача 4.** Нужно получить список клиентов с их заказами. В результат должны попасть даже те клиенты, которые ничего не заказывали. Используются таблицы Sales.Customers и Sales.Orders.

#### **4.1.3 Вложенные запросы**

**Задача 5.** Напишите запрос, который возвращает все заказы, сделанные в последний день, учтенный в таблице Orders.

**Задача 6.** Получите список стран, в которых есть клиенты, но нет сотрудников. Используются таблицы HR.Employees и Sales.Customers.

#### **4.1.4 Разные запросы**

**Задача 7.** Напишите представление, которое возвращает общий объем заказанной продукции для каждого сотрудника, разбитый по годам. Используются таблицы Sales.Orders и Sales.OrderDetails.

**Задача 8.** Напишите запрос к таблице dbo.Orders, который вычисляет для каждой строки разницу между количеством товара в текущем и предыдущем, а также в текущем и следующем заказах. Используется таблица dbo.Orders.

#### **4.1.5 Добавление, обновление и удаление данных**

**Задача 9.** Добавьте в таблицу dbo.Customers строку со следующими значениями:

- custid: 100
- companyname: Юпитер
- country: Россия
- region: RU
- city: Москва

**Задача 10.** Выберите из таблицы Sales.Customers записи о всех клиентах, которые размещали заказы, и скопируйте их в таблицу dbo.Customers.

**Задача 11.** Удалите из таблицы dbo.Orders заказы, размещенные бразильскими клиентами.

#### **4.1.6 Транзакции**

**Задача 12.** Откройте в СУБД три соединения. Запустите в одном из них произвольный код для обновления строк в таблице и продемонстрируйте транзакцию.

## **4.2 Типовые теоретические вопросы:**

### **4.2.1 Основы хранения и обработки данных. Проектирование БД**

1. Дайте определения следующим понятиям: информация, информационная система, информационный объект, информационная технология.
2. Что такое база данных и каково ее место в информационной системе.
3. В чем различие между данными и метаданными.
4. Каково назначение системы управления базами данных.
5. Каким образом прикладные программы взаимодействуют базой данных.
6. Какие компоненты входят в состав банка данных
7. Для чего используется словарь данных
8. Перечислите основные категории пользователей и разработчиков баз данных. Какова их роль в функционировании банка данных.
9. Назовите особенности каждого этапа развития СУБД.
10. Какую роль в развитии технологий баз данных сыграло появление персонального компьютера.
11. Чем характеризуется современный этап развития технологии работы с данными.
12. Какие перспективные направления развития БД.
13. Каковы функции СУБД.
14. Для чего нужна журнализация в базах данных.
15. Что такое транзакция.
16. Какие вспомогательные службы предлагает СУБД.
17. Что означает целостность базы данных
18. Что представляет собой трехуровневая архитектура СУБД.
19. В чем особенность уровня внешних моделей.
20. В чем особенность концептуального уровня.
21. В чем особенность физического уровня.
22. Что называется схемой базы данных.

23. Что означает логическая и физическая независимость данных.
24. Что такое модель данных.
25. Для чего строится модель данных.
26. Какие аспекты включает в себя модель данных.
27. Перечислите классические и современные модели представления данных.
28. Укажите достоинства и недостатки иерархической модели данных
29. Как организуется физическое размещение данных в БД иерархического типа.
30. Охарактеризуйте сетевую модель данных
31. Охарактеризуйте реляционную модель данных
32. В чем особенность постреляционной модели данных
33. Где находят применение многомерные модели данных.
34. Назовите достоинства и недостатки многомерной модели данных.
35. Охарактеризуйте многомерную модель.
36. Приведите примеры многомерных таблиц
37. Назовите и объясните смысл операций, выполнимых над данными в случае многомерной модели.
38. Какими характеристиками должна обладать СУБД, чтобы называться объектно-ориентированной.
39. Назовите достоинства и недостатки объектно-ориентированной модели представления данных.
40. Раскройте смысл понятий многомерной СУБД: измерение, ячейка, срез.
41. Дайте определение основным понятиям реляционной модели данных: отношение, тип данных, домен, атрибут, кортеж, схема отношения, первичный ключ.
42. Дайте математическое определение понятиям: отношение, тип данных, домен, атрибут, кортеж, схема отношения, первичный ключ.
43. Опишите основные свойства отношений.
44. Что называется реляционной алгеброй.
45. Опишите базовые теоретико-множественные операции реляционной алгебры.

46. Опишите специальные реляционные операции.
47. Что называется унарной, а что бинарной операцией.
48. Приведите примеры объединения, пересечения, вычитания отношений.
49. Почему произведение отношений используется редко.
50. Что общего между операциями произведения и соединения.
51. В чем особенность операций эквисоединения и естественного соединения.
52. Что называется индексом в базе данных.
53. За счет чего происходит увеличение скорости работы с данными в индексированной таблице.
54. Объясните механизм связывания таблиц в базах данных.
55. Что означает понятие целостности данных
56. Опишите особенности работы со связанными таблицами.
57. Охарактеризуйте типы связей «один к одному», «один ко многим», «многие ко многим».
58. В каких случаях используется связь «один к одному»
59. Почему надо избавляться от связи «многие ко многим».
60. Перечислите достоинства и недостатки реляционной модели данных.
61. Перечислите основные этапы проектирования баз данных.
62. Для чего строится модель предметной области.
63. Что называется концептуальной моделью.
64. Какие требования предъявляются к модели предметной области.
65. Какие базовые понятия используются на этапе концептуального проектирования.
66. Какие задачи решаются на этапе концептуального проектирования.
67. Что включает в себя концептуальная модель базы данных.
68. Перечислите шаги концептуального проектирования.
69. Что составляет семантическую основу ER-модели.
70. Что называется сущностью и экземпляром сущности.
71. Что называется атрибутом сущности и экземпляром атрибута.

72. Что называется связью между сущностями.
73. Какая сущность является ассоциированной сущностью.
74. Как определить необходимую степень детализации атрибута.
75. Что означает модальность связи и как она обозначается в ER-диаграмме.
76. Что означает полная идентификация сущности
77. Что называется логической моделью данных.
78. Какие базовые понятия используются на этапе логического проектирования.
79. Какие задачи решаются на этапе логического проектирования.
80. Перечислите шаги логического проектирования.
81. На каком этапе производится оценка требований к вычислительным ресурсам, необходимым для функционирования БД.
82. Опишите типовую пошаговую процедуру преобразования диаграммы «Сущность-связь» в реляционную схему БД.
83. Для чего используется модель данных физического уровня.
84. Какие базовые понятия используются на этапе физического проектирования.
85. Перечислите шаги физического проектирования.
86. Какие задачи решаются на этапе физического проектирования
87. Назовите цели нормализации.
88. Чем опасно избыточное дублирование информации.
89. Назовите основные свойства нормальных форм.
90. Какие ограничения таблиц относят к 1НФ, 2НФ, 3НФ.
91. Приведите примеры таблиц, соответствующих и не соответствующих требованиям нормальных форм.
92. Дайте определение Case-средствам и Case-технологии.
93. Перечислите требования к перспективной Case-системе.
94. Назовите признаки классификации Case-средств.

#### **4.2.2 Разработка и администрирование БД**

1. Какие требования предъявляются к языку работы с базами данных.

2. Что называется реализацией языка SQL.
3. Почему язык SQL стал популярным среди разработчиков.
4. Какие категории имеются в SQL.
5. Какие команды относятся к каждой категории языка.
6. Какие типы данных используются в SQL.
7. Каким образом используются домены при создании таблиц.
8. Какие ограничения могут быть описаны в домене.
9. Как задается первичный ключ при создании таблиц.
10. Что называется ограничением ссылочной целостности и как оно задается в SQL.
11. Как создается связь при определении таблиц в языке SQL.
12. Опишите формат оператора Select.
13. Как сгруппировать результат запроса.
14. Как наложить ограничения на результат запроса.
15. Как отсортировать результат запроса.
16. Что называется внутренним соединением таблиц
17. В каких случаях в запросах используется предложение Having.
18. Что называется подзапросом.
19. Какой подзапрос называется скалярным, а какой – табличным.
20. Для чего в подзапросе используются предложения Singular, Exists, All, Some.
21. Что называется внешним соединением таблиц и чем внешнее соединение отличается от внутреннего соединения.
22. Что называется генератором.
23. Для чего используются хранимые процедуры.
24. В каких случаях следует создавать индексы.
25. Как осуществляется добавление новых строк в таблицу с помощью SQL
26. Как осуществляется удаление строк из таблицы с помощью SQL
27. Как осуществляется изменение строк в таблице с помощью SQL
28. Каковы преимущества использования языка SQL.

### 4.2.3 Организация защиты данных в хранилищах

1. В чем заключается разница между режимом Windows и смешанным режимом аутентификации?
2. Какая разница между регистрационным именем входа в SQL Server и учетной записью пользователя базы данных?
3. Для чего используется инструкция Grant
4. Для чего используются триггеры.
5. При каких событиях происходит вызов триггера.
6. В чем основное отличие триггера от хранимой процедуры
7. Какая цель использования транзакций?
8. В чем заключается разница между локальной и распределенной транзакцией
9. В чем заключается разница между явным и неявным режимом транзакции
10. Какие типы блокировок совместимы с монопольной блокировкой?
11. Как можно проверить, было ли успешным выполнение каждой инструкции
12. Transact-SQL?
13. В каких случаях следует использовать инструкцию SAVE TRANSACTION?
14. В чем заключается разница между блокировкой уровня строк и блокировкой уровня страниц?
15. Может ли пользователь явно влиять на реализацию блокировок системой?
16. В чем состоит разница между основными типами блокировки (разделяемой и монопольной) и блокировкой намерения?
17. Что означает понятие укрупнения блокировки?
18. Изложите разницу между уровнями изоляции READ UNCOMMITTED и SERIALIZABLE
19. Что такое взаимоблокировка?

20. Какой процесс в качестве "жертвы" в случае взаимоблокировки? Может ли пользователь повлиять на решение системы в этом вопросе?