

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное бюджетное образовательное учреждение высшего  
образования  
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий  
(наименование института полностью)

Кафедра « \_\_\_\_\_ Прикладная математика и информатика \_\_\_\_\_ »  
(наименование)

09.03.03 Прикладная информатика

(код и наименование направления подготовки, специальности)

Бизнес-информатика

(направленность (профиль) специализация)

## ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)

на тему \_\_\_\_\_ Автоматизация обучения стажеров-программистов

1С:Предприятие

Студент

\_\_\_\_\_ К.В. Колчинцева \_\_\_\_\_

(И.О. Фамилия)

\_\_\_\_\_ (личная подпись)

Руководитель

\_\_\_\_\_ к.т.н., доцент, Э.В. Егорова \_\_\_\_\_

(ученая степень, звание, И.О. Фамилия)

Тольятти 2020

## Аннотация

Выпускная квалификационная работа на тему: «Автоматизация обучения стажеров-программистов 1С:Предприятие» содержит 82 страницы текста, рисунков – 31, таблиц – 15, использованных источников – 20 и приложения – 5.

Ключевые слова: обучающая программа, реинжиниринг, 1С, 1С:Предприятие, база данных.

Объект исследования – процесс обучения стажеров в организации.

Предмет исследования – обучающая программа.

Цель – автоматизация процесса обучения для кандидатов на должность «программист» в предприятие ООО «Трио Т».

Теоретическая часть содержит основы работы с продуктами фирмы «1С», основы работы с базами данных, характеристику предприятия, обоснование необходимости разработки нового продукта при наличии существующих на рынке аналогов и составление технического задания по разработке обучающей программы.

В практической части описываются этапы создания обучающей программы, такие как разработка концептуальной модели, структуры базы данных, разработка пользовательского интерфейса и итоги тестирования итогового продукта.

В заключении подводятся итоги применения обучающей программы на практике, проводится анализ эффективности разработки, рассматривается план программы обучения, примеры задач, а также проводится доработка недочетов программы и описываются, достигнутые при проведении реинжиниринга бизнес-процессов, результаты.

## Оглавление

Введение.....	4
Глава 1 Составление технического задания по разработке обучающей программы по программированию «1С».....	7
1.1 Характеристика предприятия ООО «ТриоТ».....	7
1.2 Разработка концептуальных моделей бизнес-процесса обучения.....	10
1.3 Анализ необходимости разработки нового продукта .....	13
1.4 Изучение основ работы с продуктами фирмы «1С».....	14
1.5 Формирование технического задания по разработке обучающей программы на платформе «1С:Предприятие» .....	19
Выводы по главе 1 .....	21
Глава 2 Разработка и проектирование обучающей программы по программированию «1С» .....	22
2.1 Проектирование обучающей программы .....	22
2.2 Реализация программы в «1С» .....	34
2.3 Результаты тестирования обучающей программы.....	46
Выводы по главе 2.....	48
Глава 3 Применение обучающей программы и анализ эффективности .....	50
3.1 Заполнение теоретического материала, составление тестовых и практических заданий.....	50
3.2 Итоги обучения стажеров-программистов и внесение изменений в обучающую программу .....	53
3.3 Анализ эффективности обучения стажеров по новой модели бизнес-процесса .....	57
Выводы по главе 3.....	59
Заключение .....	61
Список используемой литературы .....	63
Приложение А Техническое задание по разработке обучающей программы ..	65
Приложение Б Шаблон плана тестирования .....	67

Приложение В Программный код проверки доступности ролей «Стажер» и «Наставник» .....	69
Приложение Г Реализация модуля формы справочника «Материалы занятий»	70
Приложение Д Реализация модуля формы задачи «Тестовое задание» .....	75

## Введение

Фирма «1С» - российская компания, специализирующаяся на дистрибуции, поддержке и разработке компьютерных программ и баз данных делового и домашнего назначения. Данная фирма ведет свою деятельность уже несколько десятков лет, и поставляемые ими программные обеспечения находят свое применение в различных сферах деятельности, связанных с ведением бизнеса: бухгалтерия, торговля, учет зарплаты и т.д. Помимо предоставляемых программных обеспечений, называемых далее «типовые решения», платформа «1С:Предприятие» позволяет производить разработку собственных решений. Возможно и изменение существующих типовых конфигураций для клиентов, имеющих потребности в решениях, не предоставляемых фирмой.

Выбранная тема «Автоматизация обучения стажеров-программистов 1С:Предприятие» направлена на создание обучающей программы и прохождения стажировки на должность программиста в предприятие ООО «Трио Т».

При выполнении выпускной квалификационной работы будет разработана обучающая программа (база данных) для обучения стажеров программистов.

Занесенный в обучающую программу теоретический и практический материал поможет автоматизировать обучение новых сотрудников: стажер будет проводить собственное обучение по материалам и выполнять тестовые задания в программе, а практические задания будут выполняться под курированием наставника.

Разработка обучающей программы по программированию и конфигурированию баз данных фирмы «1С» имеет большую актуальность:

- текущий бизнес-процесс обучения специалистов (изучение теоретического материала для выполнения итогового задания) может занимать больше установленного временного промежутка времени;

- сложно производить оценку результатов текущего бизнес-процесса обучения специалистов;
- материалы обучения работе с продуктами фирмы «1С» чаще всего платные, в ином же случае они уже утратили свою актуальность в связи с быстрым развитием продуктов;
- официальные источники информации «1С» чаще всего имеют сухую теорию без возможности решения практических заданий.

Целью выпускной квалификационной работы является автоматизация процесса обучения для кандидатов на должность «программист» в предприятие ООО «Трио Т». По окончании разработанного курса, стажер должен с легкостью сдать экзамен «1С:Профессионал», проводимые фирмой «1С» для аттестации специалистов.

Для выполнения поставленной цели ВКР требуется выполнить список задач:

- создать техническое задание на разработку обучающей программы;
- разработать обучающую программу;
- применить программу в процессе обучения стажеров;
- провести анализ эффективности применения обучающей программы.

Объектом исследования будет являться процесс обучения стажеров в организации, а предметом исследования будет выступать обучающая программа.

При выполнении выпускной квалификационной работы в рамках теоретической части будут рассмотрены основы работы с базами данных и характеристика предприятия, приведено обоснование необходимости разработки нового продукта при наличии существующих на рынке аналогов и составлено техническое задание по разработке обучающей программы.

Практическая часть будет содержать в себе этапы создания обучающей программы, такие как построение диаграмм и концептуальных моделей,

разработка структуры базы данных, пользовательского интерфейса и подведение итогов тестирования конечного продукта.

В заключении производится применение обучающей программы на практике, анализируются временные затраты на обучение, рассматривается план программы обучения, примеры задач, а также проводится доработка недочетов программы и описываются, достигнутые при проведении реинжиниринга бизнес-процессов, результаты.

# Глава 1 Составление технического задания по разработке обучающей программы по программированию «1С»

## 1.1 Характеристика предприятия ООО «ТриоТ»

Перед началом создания обучающей программы требуется провести анализ предприятия, для которого будет производиться автоматизация обучения кандидатов, выявить недостатки бизнес-процесса «как есть сейчас» обучения и составить вариант бизнес-процесса «как должно быть».

Организация ООО «ТриоТ» является коммерческой, распространяющая информационную систему «1С» и осуществляющая поддержку и сопровождение данного продукта. Тип организации – предпринимательский.

При ведении разработок компания использует методологию внедрения OneMethodology, детально описанную В.И. Грекул в учебнике «Управление внедрением информационных систем». [13]

На предприятии имеются несколько условных отделов разбитых по специальностям работников. Структурная схема представлена на рис. 1.1.

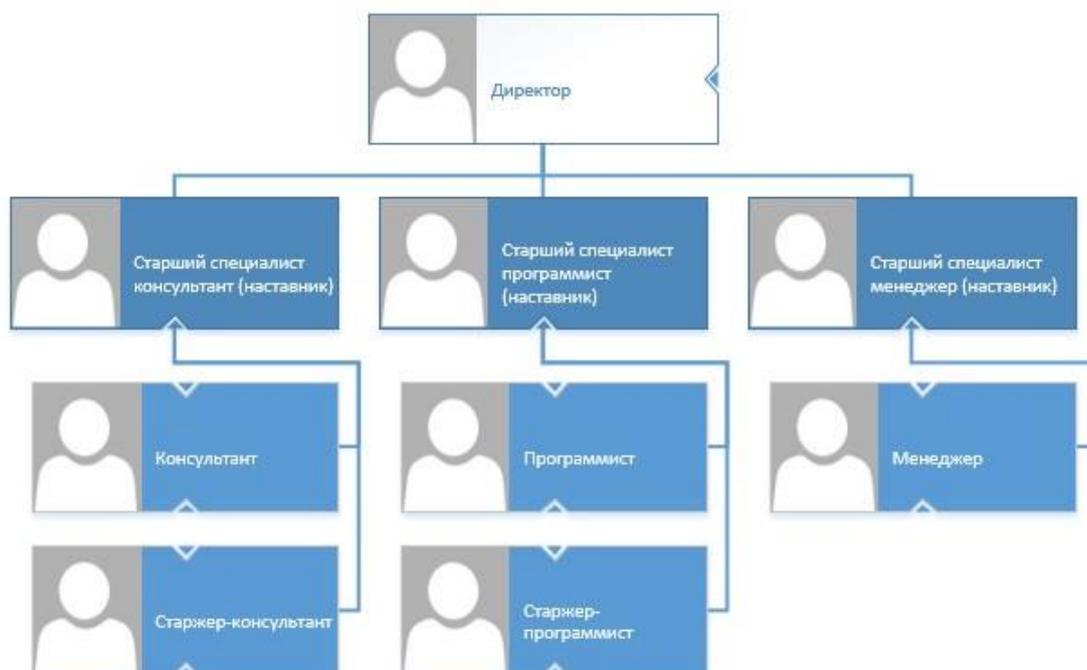


Рисунок 1.1 – Структурная схема ООО «Трио Т»

Среди представленных должностей рассмотрим более углублено должности «старший специалист программист – «наставник» и «стажер-программист».

Обязанностями должности «старший специалист программист (наставник)» являются выполнение текущих задач компании по разработке/доработке программного функционала и работа с кандидатами на должность «программист». Стажеры же выполняют только практическое задание параллельно самостоятельно изучая материал, предоставленный организацией.

Организация ООО «Трио Т» предоставляет каждому работнику персональный компьютер для работы. Каждый компьютер имеет процессор со встроенным графическим ядром не меньше 2017 года выпуска, а также оперативную память минимум 8ГБ для быстрых работ с базами данных.

В предприятии имеются отдельно выделенные сервера для работы с большими клиентами. Примеры технического обеспечения можно увидеть в таблице 1.1.

Из программного обеспечения организация использует поставляемые продукты фирмы «1С». Если требуется добавление недостающего функционала организация производит собственные доработка с помощью сотрудников предприятия. Примеры программного обеспечения можно увидеть в таблице 1.1. [11]

Таблица 1.1 – Анализ аппаратного и программного обеспечения

№	Техническое/программное обеспечение	Требует обновления (Да/Нет)
Техническое обеспечение		
1	Стационарный компьютер, Процессор Intel® Pentium® CPU G4560, ОЗУ 8ГБ, SSD.	Нет
2	Серверное оборудование, Процессор Intel® Xeon® E3-1270 V2, ОЗУ 28ГБ	Нет
Программное обеспечение		
1	1С:Управление торговлей (11.4.7.150)	Нет
2	1С:CRM (3.0.15.3)	Нет

Примеры программного и технического обеспечения составлялись в классификацию основных элементов, представленных И. М. Приваловым в книге «Основы аппаратного и программного обеспечения». [11]

Подведем итог аппаратного и программного анализа: в организации техническое и программное обеспечение поддерживается на допустимом уровне работы.

## 1.2 Разработка концептуальных моделей бизнес-процесса обучения

Произведем анализ предметной области с использованием CASE-средства. Важно усовершенствовать бизнес-процессы, перед тем как их автоматизировать. Причины необходимости усовершенствования приводит Дж. Джестон в своей книге «Управление бизнес-процессам: практ. руководство по успешной реализации проектов». [3]

Понятие «бизнес-процесс» детально раскрывает А.В. Варзунов в книге «Анализ и управление бизнес-процессами». [6]

Для начала опишем контекстную модель исследуемого бизнес-процесса «КАК ЕСТЬ» (AS IS), представленную на рис. 1.2.



Рисунок 1.2 – Концептуальная модель предметной области

На представленной схеме мы видим процесс приема кандидатов на должность «программист». Для этого кандидат проводит самостоятельное изучения сферы деятельности предприятия и выполняет практическое задание. По выполнению задания проводится проверка выполненной работы и выводится решение о приеме на работу кандидата.

Произведем декомпозицию процесса, согласно правилам выделения процессов, описанные В.Г. Елиферовым в учебнике «Бизнес-процессы: регламентация и управление». [1]

Декомпозиция процесса представлена на рис. 1.3.

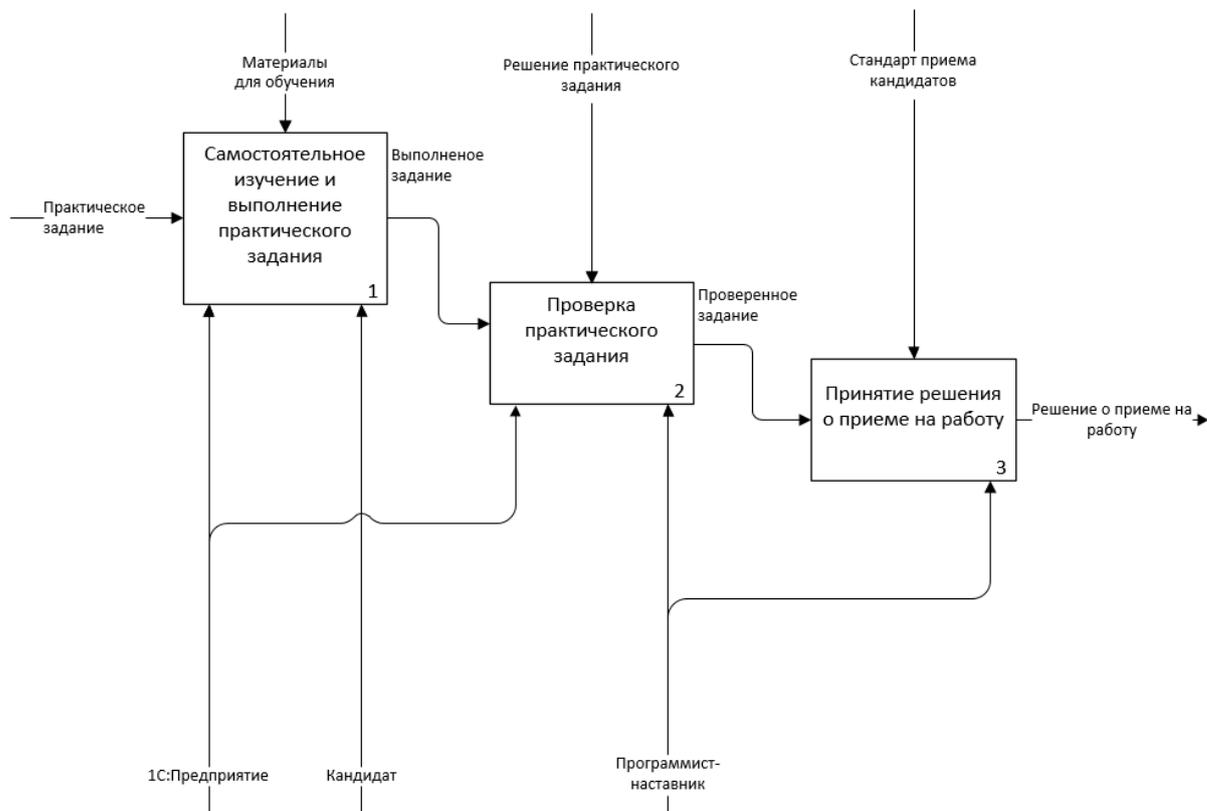


Рисунок 1.3 – Декомпозиция концептуальной модели предметной области

Такой процесс не оптимален. Во-первых, кандидат производит самостоятельное изучение по множеству неструктурированного материала. Во-вторых, у кандидата нет уточнить непонятные ему моменты до сдачи задания. В-третьих, для полного усвоения предоставленных материалов одного практического задания может быть мало. [12]

Сделаем вывод: для правильного обучения кандидатов на должность «программист» требуется добавление обучающей программы, которая будет включать в себя как теоретический материалы, так и практический для каждой выделенной главы, а также требуется курирование обучения со стороны наставника.

Используя основные принципы и правила реинжиниринга, описанные А.О. Блиновым в учебном пособии «Реинжиниринг бизнес-процессов», произведем реинжиниринг бизнес-процесса обучения. [5]

Концептуальная модель для вида «ТО-ВЕ» представлена на рис 1.4.

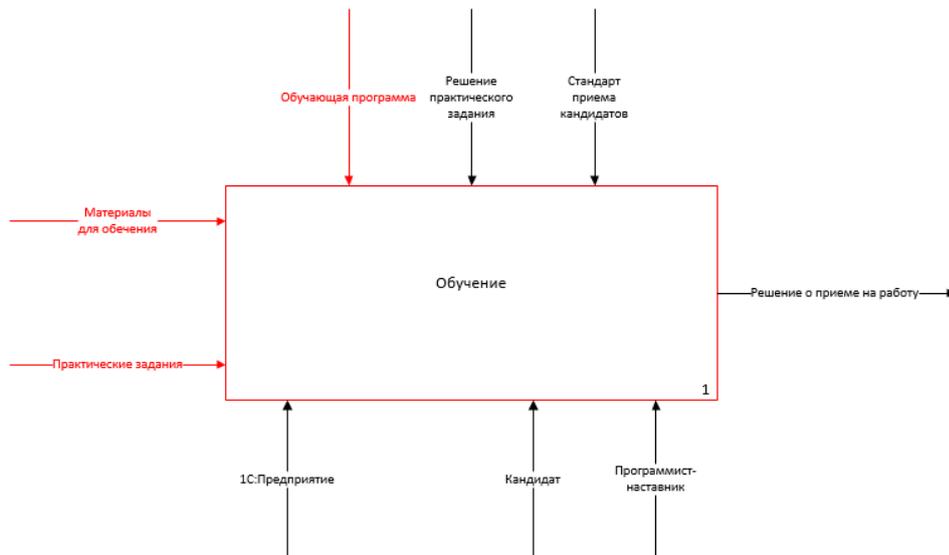


Рисунок 1.4 – Концептуальная модель предметной области «ТО-ВЕ»

Произведем декомпозицию процесса «ТО-ВЕ», представленную на рис.

1.5.

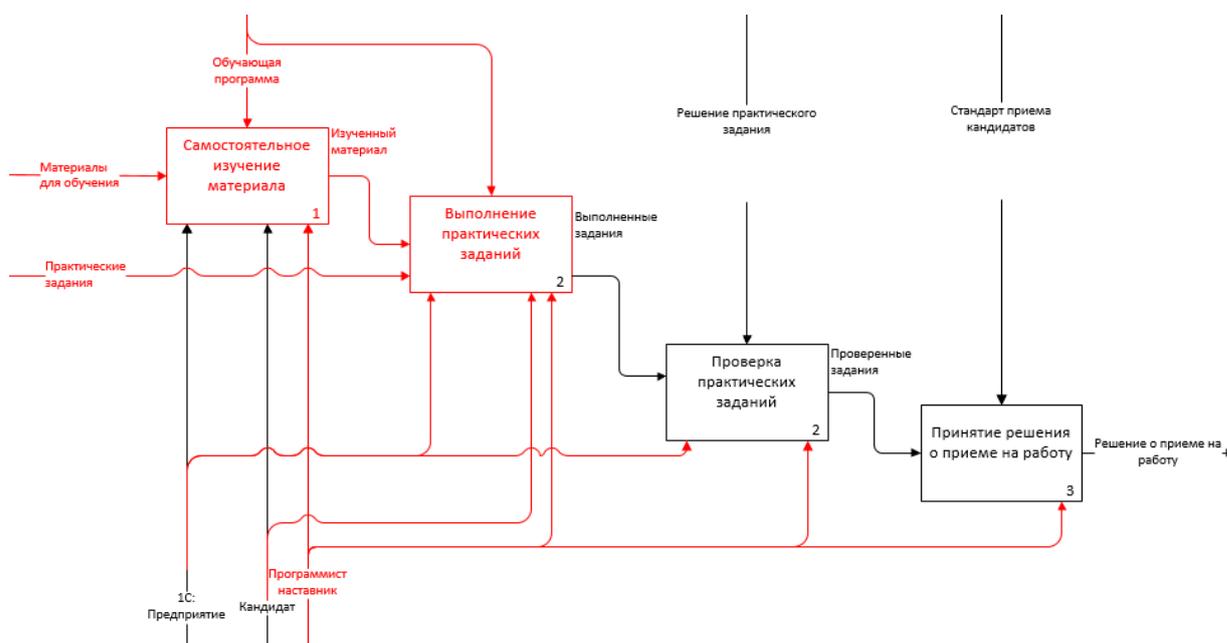


Рисунок 1.5 – Декомпозиция концептуальной модели предметной области «ТО-ВЕ»

Красным цветом выделены изменения, внесенные в модель бизнес-процесса. Для модели «как должно быть» мы видим, что наставник участвует как в процессе обучения, так и в выполнении задания. Он помогает стажеру со всеми непонятными ему вопросами оперативно. Обучающая программа призвана контролировать проведение обучения и структурировать его.

Также, происходит разделение процесса изучения материала и выполнения практического задания, что позволяет сначала получить требуемые знания и после их закрепить, а не производить поиск нужной информации по конкретно поставленной задаче.

### **1.3 Анализ необходимости разработки нового продукта**

В современном мире существует множество готовых решений, способных выполнить установленные организацией требования.

Рассмотрим список обязательных требований к обучающей программе:

- поддержка создания отчетов по успеваемости;
- возможность работы нескольких пользователей одновременно;
- возможность добавления материалов занятий, тестовых и практических заданий;
- возможность установки доступа к материалам занятий;
- возможность проверки заданий со стороны наставника;
- хранение всех материалов для обучения в едином месте;
- возможность хранения файлов размером более 100 Мб;
- использование программных продуктов фирмы «1С»;
- возможность дальнейшей поддержки проекта и добавления нового функционала;

Одними из самых ограничивающих требований является использование продуктов фирмы «1С» и возможность поддержки проекта.

Организация предполагает дальнейшее расширение обучающей программы и добавление нового функционала.

Большинство обучающих программ от фирмы «1С» ориентированы на ведение крупных курсов, а также они не поддерживают возможности выполнения практических заданий и возможности проверки заданий со стороны наставника. [8]

Из этого следует, что разработка нового программного продукта обоснована.

#### **1.4 Изучение основ работы с продуктами фирмы «1С»**

Так как требуется создать обучающую программу на базе платформы «1С», рассмотрим основы работы с продуктами данной компании. Понимание базовых основ позволит избежать ошибок при разработке обучающей программы, а также подбора материала и составлении тестовых и практических заданий. [10]

Продукты фирмы «1С» позволяют автоматизировать экономическую и организационную деятельности предприятия. Поскольку такая деятельность может быть довольно разнообразной, компания предлагает различные прикладные решения под нужды крупных сфер деятельности человека.

Одним из основных программных продуктов фирмы «1С» является технологическая платформа «1С:Предприятие». Это базовая программная среда, в которой выполняются прикладные конфигурации. Все программные решения и краткие руководства к ним представлены на официальном сайте «1С:ИТС». [4]

«1С:Предприятие» является СУБД. Система управления базами данных (СУБД) — специализированная программа (чаще комплекс программ), предназначенная для организации и ведения базы данных. Современная классификация СУБД предусматривается реализацию иерархических, сетевых и реализационных моделей данных, описание которых раскрывает Т.

М. Татарникова в своей книге «Системы управления базами данных: учебное пособие». [2]

Рассмотрев понятие «СУБД», нельзя не затронуть понятие «база данных». База данных – представленная в объективной форме совокупность самостоятельных материалов (статей, расчётов, нормативных актов, судебных решений и иных подобных материалов), систематизированных таким образом, чтобы эти материалы могли быть найдены и обработаны с помощью электронной вычислительной машины (ЭВМ).

Одной из основных классификаций баз данных является классификация по модели данных, детально описанные N. Sharma в книге «Database fundamentals (2010)». [18]

Платформа имеет свой развитый язык программирования и полностью определяет функциональные возможности решаемых задач.

Но наличие одной лишь платформы не позволяет работать с базами данных в полной мере, для этого требуется приобретение или разработка конфигураций. Конфигурации «1С» (или прикладные решения «1С») – это программы, предназначенные для автоматизации деятельности различных организаций и частных лиц.

Существует одна платформа («1С:Предприятие») и множество конфигураций. Для функционирования какого-либо прикладного решения всегда необходима платформа и какая-либо (одна) конфигурация (рис. 1.6).



Рисунок 1.6 – Конфигураций много, а платформа одна

Сама по себе платформа не может выполнить никаких задач автоматизации, так как она создана для обеспечения работы какой-либо

конфигурации. То же самое с конфигурацией: чтобы выполнить те задачи, для которых она создана, необходимо наличие платформы, управляющей ее работой.

Технологическая платформа «1С:Предприятие» имеет различные режимы работы:

- режим «1С:Предприятие» является основным и служит для работы пользователей системы. В этом режиме пользователи вносят данные, обрабатывают их и получают итоговые результаты;
- режим «Конфигуратор» предназначен для разработчиков и администраторов информационных баз. Этот режим и предоставляет инструменты, необходимые для модификации существующей или создания новой конфигурации.

Любая информационная база внутренне разделяется две или более конфигурации: конфигурация, предназначенная для разработчика, называется «основная конфигурация», а конфигурация, с которой работают пользователи, называется «конфигурация базы данных». Основная конфигурация имеет возможность редактирования – «конфигурируемость». Для конфигурации базы данных доступно только обновление на основе конфигурации для разработчиков.

Существует и третий вид конфигурации – это «конфигурация поставщика». Данный вид присутствует только в поставляемых прикладных решениях, но в которые были внесены изменения сторонними разработчиками.

Внутреннее разделение на несколько конфигурации позволяет вносить изменения, не прерывая работы пользователей. Если эти изменения не затрагивают структуру базы данных, то обновить конфигурацию базы данных можно не прерывая работы пользователей. Это так называемое «динамическое обновление». Пользователи увидят изменения только после того, как перезапустят свое приложение.

Для продуктов фирмы «1С» существует два вида хранения информационных баз: файловая и клиент-серверная.

Файловый вариант работы предназначен для небольшого количества пользователей. База данных такого варианта состоит из одного файла и все операции или запросы (проведение документов, формирование отчетов, поиск документов, закрытие периода и пр.) выполняются непосредственно на компьютере пользователя. Такой способ требует производительного оборудования для каждого работающего пользователя в системе. Пример работы файлового режима представлен на рис. 1.7.

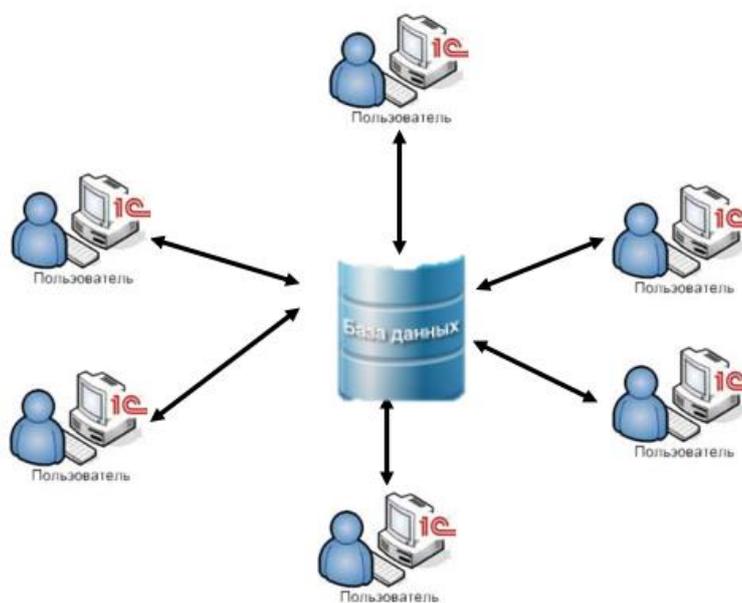


Рисунок 1.7 – Схема работы в файловом режиме.

Серверный режим работы включает в себя иную структуру работы с базой данных, представленную на рис. 1.7 и требует установленной программы «1С:Сервер».



Рисунок 1.7 – Схема работы в клиент-серверном режиме.

В отличие от файлового режима в клиент-серверном база данных состоит не из одного файла, а из множества разных файлов.

Более ресурсоемкие операции выполняются на сервере, в месте хранения файлов базы данных, и не нагружают персональный компьютер пользователя. Задача компьютеров пользователей – отражение визуальной оболочки программы.

Поскольку задачи автоматизации могут быть самыми разными, фирма «1С» и ее партнеры выпускают прикладные решения, каждое из которых предназначено для автоматизации одной определенной области человеческой деятельности. В качестве примера существующих прикладных решений можно перечислить следующие типовые решения:

- «1С:Бухгалтерия»;
- «1С:Управление торговлей»;
- «1С:Зарплата и управление персоналом»;
- «1С:Розница».

Существует также множество других типовых прикладных решений. Более подробно о них можно узнать на официальном сайте фирмы «1С».

Типовое прикладное решение является, универсальным и способно удовлетворить потребности самых разных предприятий, работающих в одной области деятельности. Такая универсальность неизбежно приводит к тому, что на конкретном предприятии будут использоваться далеко не все возможности прикладного решения, а каких-то возможностей в нем будет недоставать.

Данная проблема исключается с помощью возможности конфигурируемости системы. Платформа, помимо управления работой конфигурации, содержит средства, позволяющие вносить изменения в используемую конфигурацию. Более того, платформа позволяет создать свою собственную конфигурацию с нуля, если по каким-либо причинам использование типовой конфигурации представляется нецелесообразным.

Добавить новый функционал без внесения изменений в конфигурацию также представляется возможным. Это реализуется с помощью типового механизма «внешних обработок».

Внешние обработки представляют собой отдельные файлы с расширением \*.erf, которые не входят в состав прикладного решения.

Примеры разработок внешних объектов представил А.П. Габец в пособии «1С: Предприятие 8.2. Простые примеры разработки». [9]

Кроме того, важным преимуществом внешних обработок является возможность проектировать и отлаживать их в процессе работы прикладного решения, без необходимости сохранения каждый раз конфигурации прикладного решения.

Некоторые внешние обработки, созданные сторонними разработчиками, можно найти в бесплатном доступе. Одной из таких является «Инструменты разработчика». Данная обработка имеет огромный функционал в своем составе, например, возможность редактирования элементов базы данных без выполнения проверок или выполнение запросов в режиме «1С:Предприятие». Использование инструментов для программистов «1С» не обязательно, но оно значительно упрощает работу.

Еще одним помощником для программистов является специально разработанная типовая конфигурация «1С:Библиотека стандартных подсистем» или просто «БСП». БСП — это инструментарий, предназначенный для разработчика прикладных решений. Она содержит набор универсальных функциональных подсистем, которые могут быть использованы в разрабатываемой конфигурации как все вместе, так и по отдельности. При выполнении больших разработок программисты «1С» часто используют БСП для сокращения времени на разработку.

## **1.5 Формирование технического задания по разработке обучающей программы на платформе «1С:Предприятие»**

И так, определившись с причинами необходимости разработки обучающей программы, а также изучив основы работы с продуктами «1С» и основы работы с базами данных мы можем составить техническое задание по разработке обучающей программы.

Техническое задание будет делиться на две основных части:

- требования к обучающей программе;
- план работы.

Список требований к обучающей программе описывает функционал, обязательный для данного программного продукта.

Рассмотрим требования к обучающей программе:

1. работа на платформе «1С: Предприятие 8.3» (8.3.15.18.30);
2. поддержка типового функционала (варианты отчетов, дополнительные отчеты и обработки и т.д.);
3. поддержка работы нескольких пользователей одновременно;
4. возможность работы наставника (полный доступ к базе данных и возможность давать доступ к курсам для стажеров);
5. возможность работы стажеров (минимальные права, чтение материалов для обучения, выполнение практических и тестовых заданий);
6. поддержка хранения выгрузок конфигураций баз данных для проверки практических заданий.

Определив требования, составим этапы плана работ:

1. Разработать концептуальную модель базы данных обучения;
2. Разработка обучающей программы:
  - a. разработать структуру базы данных;
  - b. разработать пользовательский интерфейс;
  - c. разработать отчет для проверки успеваемости кандидатов.
3. Провести предварительное тестирования и доработку обучающей программы;
4. Применение обучающей программы на практике;

План работ должен выполняться согласно календарному плану. [14]  
[15]

Полное техническое задание с календарным планом представлено в Приложении А.

### **Выводы по главе 1**

При составление технического задания по разработке обучающей программы по программированию «1С» была составлена характеристика предприятия, проведен реинжинеринг бизнес-процессов и сформировано техническое задание на разработку.

При проведении реинжиниринга бизнес-процесса обучения было установлено следующее:

1. модель бизнес-процесса «AS-IS» является неоптимальной, так как отсутствует качественный метод управления при изучении материалов и выполнении практических заданий;

2. для более качественного контроля требуется разработка обучающей программы.

## Глава 2 Разработка и проектирование обучающей программы по программированию «1С»

### 2.1 Проектирование обучающей программы

Для построения обучающей программы будет использована нотация UML. Язык UML — это стандарт в области объектно-ориентированного проектирования, который набирает популярность и как нотация для моделирования баз данных.

При проектировании программы обучения разработано в соответствии с техническим заданием (Приложение А):

- Диаграмма вариантов использования процесса обучения стажеров-программистов при помощи «use case diagram»;
- Диаграмма классов программного продукта при помощи «class diagram»;
- Концептуальная ER-модель, построенная по методологии Питера Чена;
- Концептуальная модель базы данных.

Диаграмма вариантов использования (англ. Use case diagram) в UML — диаграмма, отражающая отношения между актерами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

Разработанная диаграмма вариантов использования отражена на рис. 2.1.

Проведем анализ требований к обучающей программе и определимся с структурой базы данных:

- Справочник «Пользователи»;
- Справочник «Материалы занятий»;
- Справочник «Материалы тестовых заданий»;
- Справочник «Вопросы тестовых заданий»;
- Справочник «Материалы практических заданий»;

- Регистр сведений «Доступ к занятиям»;
- Задача «Практическое задание»;
- Задача «Тестовое задание».



Рисунок 2.1 – Диаграмма вариантов использования процесса обучения стажеров-программистов

Диаграмма классов (англ. Static Structure diagram) — структурная диаграмма языка моделирования UML, демонстрирующая общую структуру иерархии классов системы, их коопераций, атрибутов (полей), методов, интерфейсов и взаимосвязей между ними. Широко применяется не только для документирования и визуализации, но также для конструирования посредством прямого или обратного проектирования.

Диаграмма классов программного продукта представлена на рис. 2.2.

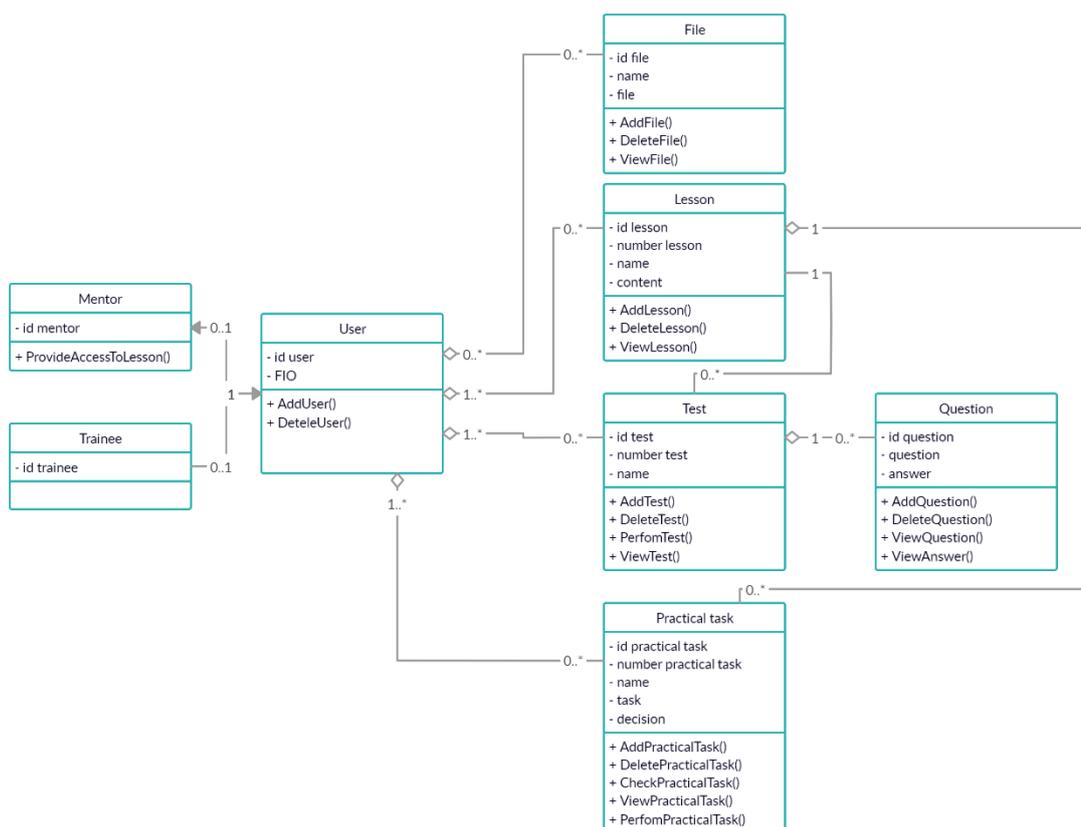


Рисунок 2.2 – Диаграмма классов программного продукта

Модель сущность-связь (ER модель для краткости) описывает взаимосвязанные вещи, представляющие интерес в определенной области знаний. Базовая модель ER состоит из типов сущностей (которые классифицируют вещи, представляющий интереса) и определяет отношения, которые могут существовать между субъектами (экземплярами этих типов сущностей).

Концептуальная ER-модель, построенная по методологии Питера Чена представлена на рис. 2.3. Использование концептуальных моделей данных высокого уровня для проектирования баз данных рассматривается более детально R. Elmasri в книге «Fundamentals of Database Systems». [17]

Разработанная концептуальная модель базы данных, представлена на рис. 2.4.

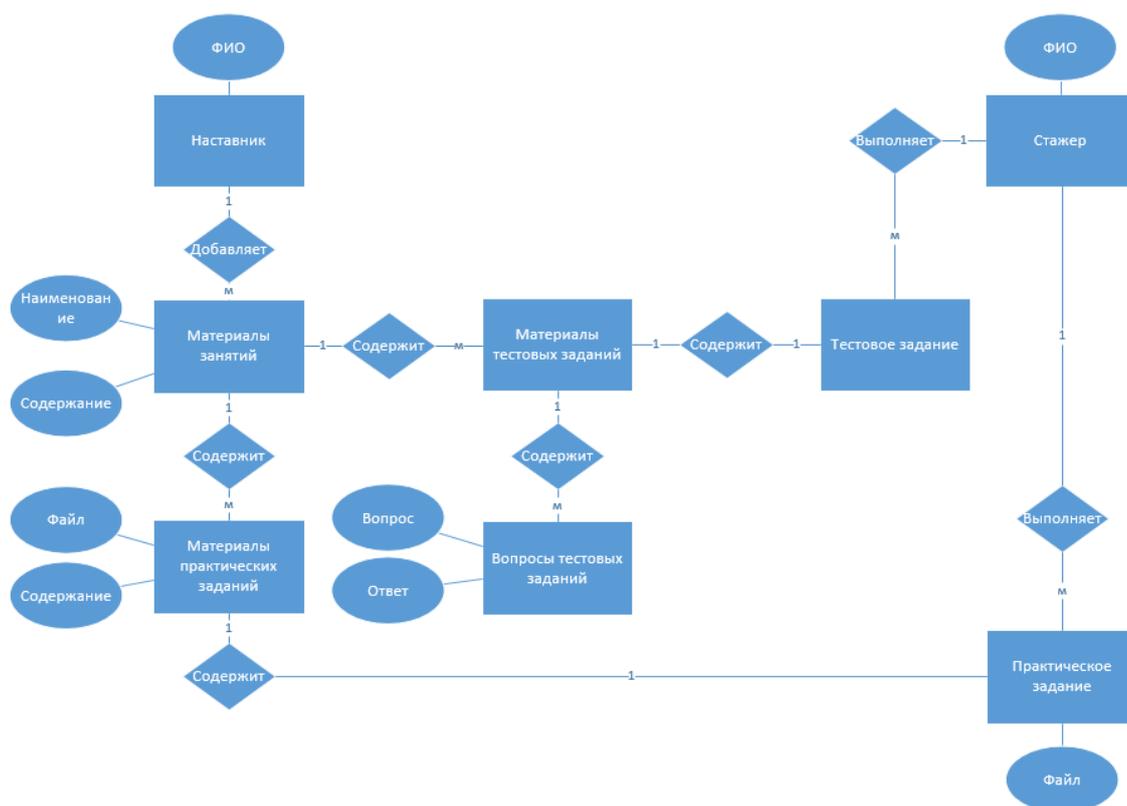


Рисунок 2.3 – Концептуальная ER-модель

Рассмотрим каждый элемент базы данных по порядку, опишем реквизиты и табличные части, доступные данному объекту конфигурации. [18] [22]

Перед началом создания требуемого функционала добавим в базу данных несколько типовых функций, описанных в техническом задании (Приложение А):

Перечень добавляемых опций:

- Варианты отчетов;
- Длительные операции;
- Дополнительные отчеты и обработки;
- Журнал регистрации;
- Пользователи;
- Резервное копирование;
- Удаление помеченных объектов;
- Файлы.

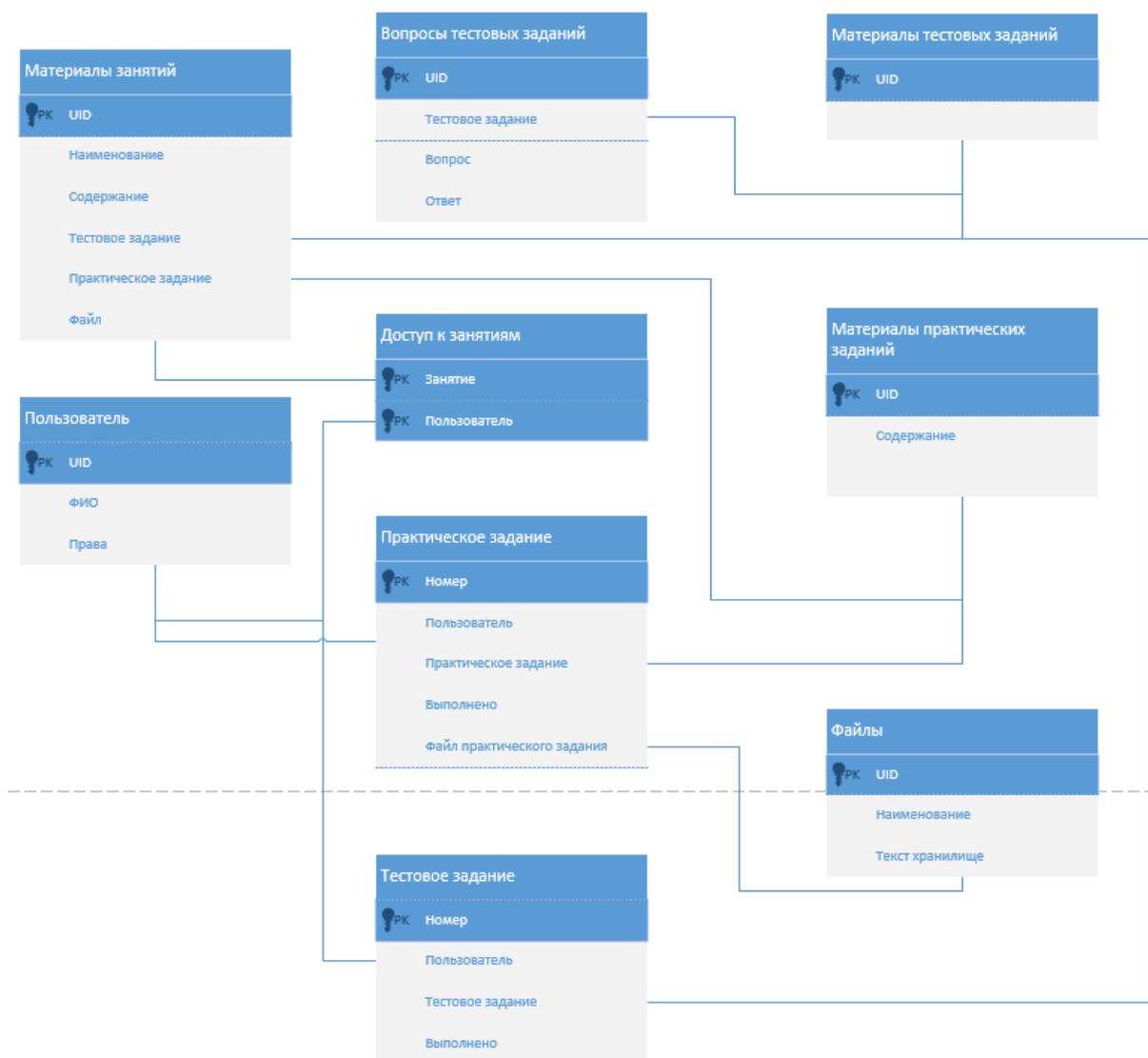


Рисунок 2.4 – Концептуальная модели базы данных

Выбранные опции присутствует во многих типовых конфигурациях, потому будет целесообразно воспользоваться для их добавления конфигурацией «1С: Библиотека стандартных подсистем». Для разработки обучающей программы будем использовать версию «Библиотека стандартных подсистем», редакция 3.0 (3.0.3.36). Применение данного функционала значительно сокращает время разработки конечного продукта и позволяет сторонним разработчикам быстрее ориентироваться в созданном решении.

Выполним доработку справочника «Пользователи» и «Файлы». Объект конфигурации «Справочник» предназначен для работы со списками данных.

Как правило, в работе любой фирмы используются списки сотрудников, списки товаров, списки клиентов, поставщиков и т. д. Таким объектом конфигурации будет являться элемент концептуальной модели «Пользователи». Опишем требуемые данные справочника «Пользователи», представленные в таблице 2.1.

Для описания структуры базы данных используем рекомендации, представленные С. Coronel в книге «Database Systems: Design, Implementation, & Management (2019)» и М. J. Hernandez в книге «Database Design for Mere Mortals: A Hands-On Guide to Relational Database Design (2013)». [16] [19]

Таблица 2.1 – Данные справочника «Пользователи»

Реквизиты и табличные части	Тип реквизита	Описание реквизита
Наименование	Строка (100)	Обязательный реквизит.
Комментарий	Строка	
Недействителен	Булево	
Идентификатор Пользователя ИБ	Уникальный Идентификатор	УИД пользователя информационной базы, с которым сопоставлен этот элемент справочника.
Идентификатор Пользователя Сервиса	Уникальный Идентификатор	УИД пользователя сервиса, с которым сопоставлен этот элемент справочника.
Свойства Пользователя ИБ	Хранилище значений	Службное хранилище некоторых свойств пользователя ИБ

Добавить сам справочник «Пользователи» из БСП будет недостаточно для возможности входа под конкретным пользователем. В первую очередь, требуется добавить несколько параметров сеанса для корректной работы справочника и базы данных в целом и заполнение в модуле сеанса.

Добавляемые параметры сеанса указаны в таблице 2.2.

Таблица 2.2 – Добавляемые параметры сеанса

Параметр сеанса	Тип параметра	Описание параметра
Выполняется запись объекта	Булево	

Выполняется обновление ИБ	Булево	
Параметры клиента на сервере	Фиксированное соответствие	
Параметры обработчика обновления	Фиксированная структура	
Текущий пользователь	Справочник. Пользователи	

Данные справочника «Файлы», представленные в таблице 2.3.

Таблица 2.3 – Данные справочника «Файлы»

Реквизиты и табличные части	Тип реквизита	Описание реквизита
Наименование	Строка (150)	Обязательный реквизит.
Полное наименование	Строка(150)	
Код	Строка(11)	Обязательный реквизит.
Описание	Строка	
Владелец файла	Справочник. Материалы занятий; Задача. Практическое задание	
Автор	Справочник. Пользователи	
Текст хранилище	Хранилище значений	Реквизит хранения загружаемого файла.

После доработки объектов БСП, приступим к работе с объектами конфигурации, которые отсутствуют в данной типовой конфигурации.

Первым для разработки будет справочник «Материалы занятий». Структура справочника представлена в таблице 2.4.

Таблица 2.4 – Структура справочника «Материалы занятий»

Реквизиты и табличные части	Формат реквизита	Описание реквизита
Наименование	Строка (100)	Обязательный реквизит.
Код	Строка(15)	Обязательный реквизит.
Содержание	Строка	Текст материала для изучения.
Дополнительная информация	Строка	Указываются ссылки на иные ресурсы информации.
Табличная часть «Тестовые задания»		
Тестовое задание	Справочник. Материалы тестовых заданий	
Табличная часть «Практические задания»		
Практическое	Справочник.	

задание	Материалы практических заданий	
Табличная часть «Файлы»		
Файл	Справочник. Файлы	Файлы содержащие материалы занятий.

В данной структуре поля «Тестовое задание», «Практическое задание» и «Файл», отраженные в концептуальной модели, вынесены в табличные части. Аналогичным способом будет установлена и связь для справочников «Материалы тестовых заданий» и «Вопросы тестовых заданий», где элементы справочника «Вопросы тестовых заданий» будут размещаться в табличной части элементов справочника «Материалы тестовых заданий». Такой подход позволит создавать несколько тестовых и практических заданий для одного урока.

Также в таблице не указываются данные UID, так как платформа «1С:Предприятие» формирует их автоматически.

Разработаем подобным образом структуры для справочников «Материалы тестовых заданий» (таблица 2.5), «Вопросы тестовых заданий» (таблица 2.6) и «Материалы практических заданий» (2.7).

Таблица 2.5 – Структура справочника «Материалы тестовых заданий»

Реквизиты и табличные части	Формат реквизита	Описание реквизита
Наименование	Строка (100)	Обязательный реквизит.
Код	Строка(15)	Обязательный реквизит.
Табличная часть «Вопросы»		
Вопрос	Справочник. Вопросы тестовых заданий	

Таблица 2.6 – Структура справочника «Вопросы тестовых заданий»

Реквизиты и табличные части	Формат реквизита	Описание реквизита
Наименование	Строка (100)	Обязательный реквизит.
Код	Строка(15)	Обязательный реквизит.
Текст вопроса	Строка(100)	
Ответ	Строка(100)	

Таблица 2.7 – Структура справочника «Материалы практических заданий»

Реквизиты и табличные части	Формат реквизита	Описание реквизита
Наименование	Строка (100)	Обязательный реквизит.
Код	Строка(15)	Обязательный реквизит.
Содержание	Строка	Указывается текст задания.

Для элементов концептуальной модели «Тестовое задание» и «Практическое задание» требуется взять иной вид объектов конфигурации – «Задачи». Они предназначены для учета заданий и описывает способ их распределения по исполнителям, с учетом организационной структуры предприятия. Адресация заданий сотрудникам определяется реквизитами, в которых можно предусмотреть многомерную ролевую маршрутизацию, например, по ролям, рабочим группам, подразделениям, помещениям, филиалам и т. д.

Отразим структуру задачи «Практическое задание» в таблице 2.8.

Таблица 2.8 – Структура задачи «Практическое задание»

Реквизиты и табличные части	Формат реквизита	Описание реквизита
Номер	Строка (9)	Обязательный реквизит.
Дата	Дата и время	Обязательный реквизит.
Кандидат	Справочник. Пользователи	Пользователь программы.
Наставник	Справочник. Пользователи	Пользователь программы.
Практическое задание	Справочник. Материалы практических заданий	
Выполнена	Булево	Отражает, что данная задача уже выполнена.
Требуется проверка	Булево	
Файл выгрузки	Справочник. Файлы	Кандидат загружает выгрузку базы в обучающую программу.
Комментарий наставника	Строка	

Структура задания «Тестовое задание» будет так же являться задачей, иметь реквизиты «Дата», «Номер», «Кандидат» и «Наставник», а также табличную часть «Вопросы», отражающую список вопросов тестирования (таблица 2.9).

Таблица 2.9 – Структура задачи «Тестовое задание»

Реквизиты и табличные части	Формат реквизита	Описание реквизита
Номер	Строка (9)	Обязательный реквизит.
Дата	Дата и время	Обязательный реквизит.
Кандидат	Справочник. Пользователи	Пользователь программы.
Наставник	Справочник. Пользователи	Пользователь программы.
Тестовое задание	Справочник. Материалы тестовых заданий	
Выполнена	Булево	Отражает, что данная задача уже выполнена.
Табличная часть «Вопросы»		
Вопрос	Справочник. Вопросы тестовых заданий	Ссылка на элемент справочника. Табличная часть заполняется автоматически, согласно данным реквизита «тестовое задание».
Ответ	Строка(100)	Ответ пользователя на вопрос, сравнивается с реквизитом «Ответ».

Еще одним объектом конфигурации является «Доступ к занятиям». Доступ к занятиям должен использоваться как справочная информация, потому подойдет объект конфигурации «Регистр сведений». Данный регистр сведений требуется для возможности отображения доступных уроков стажерам. Объект конфигурации «Регистр сведений» предназначен для описания структуры хранения данных в разрезе нескольких измерений и может хранить любые данные (а не только числовые, как регистр накопления).

Структура регистра сведений «Доступ к занятиям» отражена в таблице 2.10.

Таблица 2.10 – Структура регистра сведений «Доступ к занятиям»

Реквизиты и табличные части	Формат реквизита	Описание реквизита
Кандидат	Справочник. Пользователи	Измерение.
Занятие	Справочник. Материалы занятий	Измерение.
Наставник	Справочник. Пользователи	Ресурс.

Последним объектом конфигурации станет отчет «Успеваемость кандидатов». Данный объект не был внесен в концептуальную модель базы

данных, так как является лишь инструментом для отображения уже имеющихся данных в базе. Объект конфигурации «Отчет» предназначен для описания алгоритмов, при помощи которых пользователь сможет получать необходимые ему выходные данные. Алгоритм формирования выходных данных описывается при помощи визуальных средств или с использованием встроенного языка. В реальной жизни объектам конфигурации «Отчет» соответствуют всевозможные таблицы выходных данных, сводных данных, диаграммы и пр.

Для создания любого отчета потребуется создать «схему компоновки данных». Схема компоновки данных (СКД) – удобный конструктор для создания в программных продуктах «1С:Предприятие» сложных отчетов, способствующих развитию и отслеживанию автоматизации производства, позволяющих сделать их максимально гибкими и красивыми за минимум времени.

Отчет по успеваемости должен отражать пройденные тестовые и практические задания в разрезе конкретного кандидата.

Составим макет отчета с помощью электронных таблиц MS Excel (рис 2.5).

	A	B	C	D	E
1	Кандидат				
2		Задание	Занятие	Выполнено	Требуется проверка
3	Петров				
4		Тестовое задание 1	Занятие 1	Да	
5		Тестовое задание 2	Занятие 2	Да	
6		Тестовое задание 3	Занятие 3		
7		Практическое задание 1.1	Занятие 1	Да	
8		Практическое задание 1.2	Занятие 1	Да	
9		Практическое задание 2	Занятие 2	Да	
10		Практическое задание 3.1	Занятие 3		Да
11		Практическое задание 3.2	Занятие 3		
12	Иванов				
13		Тестовое задание 1	Занятие 1	Да	
14		Тестовое задание 2	Занятие 2	Да	
15		Практическое задание 1.1	Занятие 1	Да	
16		Практическое задание 1.2	Занятие 1	Да	
17		Практическое задание 2	Занятие 2		Да

Рисунок 2.5 – Макет отчета «Отчет по успеваемости» в MS Excel

Схема компоновки данных отчета «Отчет по успеваемости» представлена на рис 2.6.

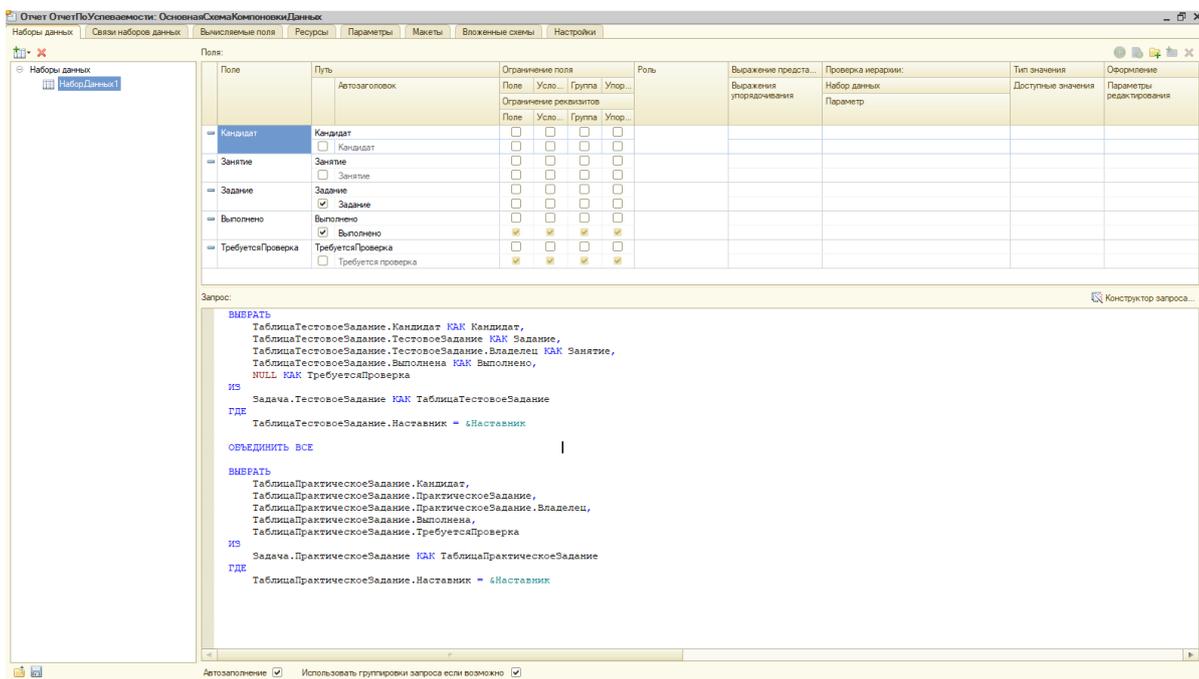


Рисунок 2.6 – Схема компоновки данных отчета «Отчет по успеваемости» в конфигураторе

Вариант отчета «Отчет по успеваемости» представлена на рис 2.7.

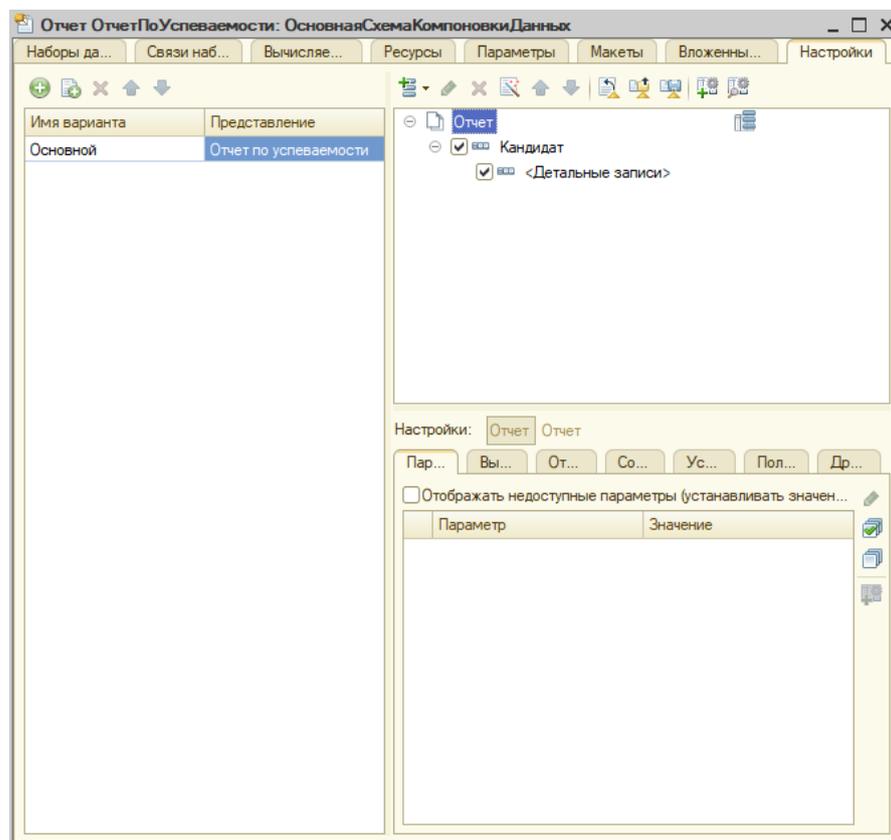


Рисунок 2.7 – Вариант отчета «Отчет по успеваемости» в конфигураторе

Выполнив проектирование обучающей программы можно приступить к реализации в «1С».

## 2.2 Реализация программы в «1С»

Одним из важных этапов разработки программных обеспечений является организация работы с пользователем. Удобство использования, пригодность использования, эргономичность — способность продукта быть понимаемым, изучаемым, используемым и привлекательным для пользователя в заданных условиях часто ставят на приоритетные места при разработке программных решений.

При разработке можно проводить косвенную оценку, которая рассматривается в стандарте ISO/IEC 25010. Косвенная оценка описывает следующие подхарактеристики удобства использования:

- **определимость пригодности:** возможность пользователя понять, подходит ли продукт или система для его потребностей, на основе первоначальных впечатлений, документации и другой предоставленной информации;
- **изучаемость:** степень эффективности, производительности и удовлетворённости пользователя обучением использованию системы;
- **управляемость:** обеспечение простоты управления и контроля;
- **защищённость от ошибок пользователя:** степень, в которой система защищает пользователя от совершения ошибок;
- **эстетика пользовательского интерфейса:** степень, в которой пользовательский интерфейс удовлетворяет пользователя и доставляет ему удовольствие от процесса взаимодействия;
- **доступность:** возможность использования продукта или системы широким кругом людей с самыми разными (в том числе, ограниченными) возможностями.

Регламентируя данной качественной оценкой и разработав все объекты конфигурации базы данных можно перейти к внешнему оформлению. Для этого потребуется разработать несколько формы для справочников и задач, а также для разделения пользователей на категории «Наставник» и «Стажер», согласно техническому заданию (Приложение А), требуется завести роли. С помощью объекта конфигурации «Роль» разработчик получает возможность описать набор прав на выполнение тех или иных действий над каждым объектом базы данных и над всей конфигурацией в целом.

Как правило, роли создаются отдельно для каждого вида деятельности, и каждому пользователю системы ставится в соответствие одна или несколько ролей.

Если пользователю поставлено в соответствие несколько ролей, предоставление доступа будет осуществляться по следующему алгоритму:

- если хотя бы в одной роли есть разрешение, то доступ будет открыт;

- если во всех ролях разрешение отсутствует, то доступ будет закрыт.

Добавим в конфигурацию базы данных две новых роли (стажер и наставник). Проверку на данные роли разместим в общем модуле «РМКВызовСервера». С помощью новых функций (Приложение В) проверки на роли «Стажер» и «Наставник» будет устанавливаться отображение элементов на формах.

Для написания программного кода использовался встроенный язык, описание которого фирма «1С» предоставляет в книге «1С:Предприятие 8.0. Описание встроенного языка». [14]

Формы в «1С:Предприятии» предназначены для отображения и редактирования информации, содержащейся в базе данных. Формы могут принадлежать конкретным объектам конфигурации или существовать отдельно от них и использоваться всем прикладным решением в целом.

Для проектирования пользовательского интерфейса используем методы моделирования А. С. Бакановым в книге «Эргономика пользовательского интерфейса. От проектирования к моделированию человеко-компьютерного взаимодействия». [12]

При разработке форм под управляемым приложением, разработчик должен сам указывать, какой код будет выполняться на сервере, а какой на клиенте. Делается это при помощи директив компиляции.

Основы и примеры разработки таких форм представлены В.А. Ажеронок в книге «Разработка управляемого интерфейса». [15]

Директиву компиляции следует указывать перед каждой функцией или процедурой в модуле формы, в общем модуле или в модуле команды. Все директивы компиляции начинаются с символа амперсанд — «&». И от них зависит, в каком контексте будет выполняться код в процедуре или функции. Всего их пять:

- &НаКлиенте
- &НаСервере

- &НаСервереБезКонтекста
- &НаКлиентеНаСервереБезКонтекста
- &НаКлиентеНаСервере

Потому часть, отвечающая за получение данных, должна выполняться на сервере, а управление элементами формы — на клиенте.

Пользователи с правами «Наставник» и «Стажер» форму справочника «Материалы занятий» должны видеть по-разному:

- стажерам доступно только прочтение материала, команды «Выполнить тест» и «Выполнить практическое задание»;
- для наставников же должна быть доступна кнопка «Редактировать», для возможности внесения изменений в материалы занятий.

Форма справочника «Материалы занятий» будет одна и при открытии формы будет изменяться доступность элементов в зависимости от доступной для пользователя роли.

Вид формы для прав «Стажер» представлен на рис. 2.5.

Рисунок 2.5 – Вид формы справочника «Материалы занятий» для прав «Стажер»

Команды «Выполнить тест» (рис. 2.6) и «Выполнить практическое задание» (рис.2.7) располагаются на соответствующих им вкладках. Каждая команда требует выбрать строку с заданием и при нажатии на кнопку выполняется проверка на наличие выполненного/начатого задания и затем открывается форма соответствующей задачи. Если задание уже выполнено, пользователю выводится сообщение об ошибке.

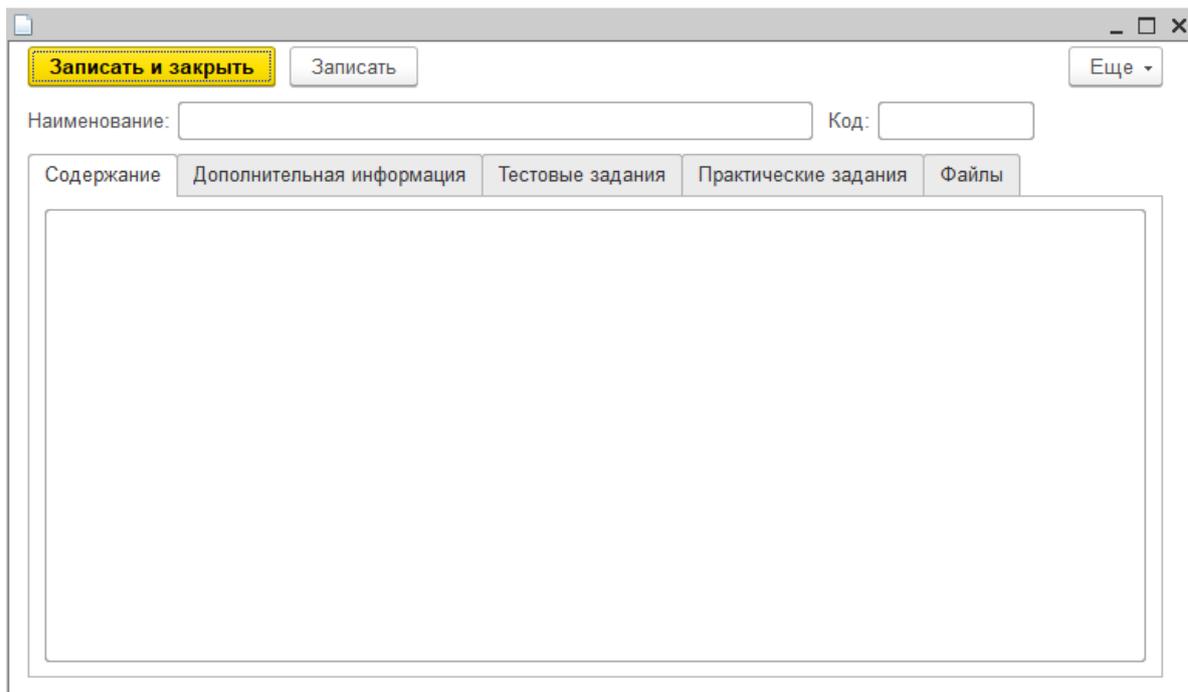
The screenshot shows a web application window titled "Занятие 1. (Материалы занятий)". At the top, there are buttons for "Записать и закрыть" (highlighted in yellow), "Записать", and "Еще". Below these are input fields for "Наименование: Занятие 1." and "Код: 000000001". A tabbed interface is visible with tabs for "Содержание", "Дополнительная информация", "Тестовые задания" (selected), "Практические задания", and "Файлы". Under the "Тестовые задания" tab, there are buttons for "Добавить", "↑", "↓", "Выполнить тест", and "Еще". Below these buttons is a table with a header row containing "N" and "Тестовое задание". The table body is currently empty.

Рисунок 2.6 – Вид формы справочника «Материалы занятий» для прав «Стажер», страница «Тестовые задания»

The screenshot shows the same web application window as Figure 2.6, but with the "Практические задания" tab selected. The buttons at the top and the input fields remain the same. The "Тестовые задания" tab is now inactive. Under the "Практические задания" tab, there are buttons for "Добавить", "↑", "↓", "Выполнить практическое задание", and "Еще". Below these buttons is a table with a header row containing "N" and "Практическое задание". The table body is currently empty.

Рисунок 2.7 – Вид формы справочника «Материалы занятий» для прав «Стажер», страница «Практические задания».

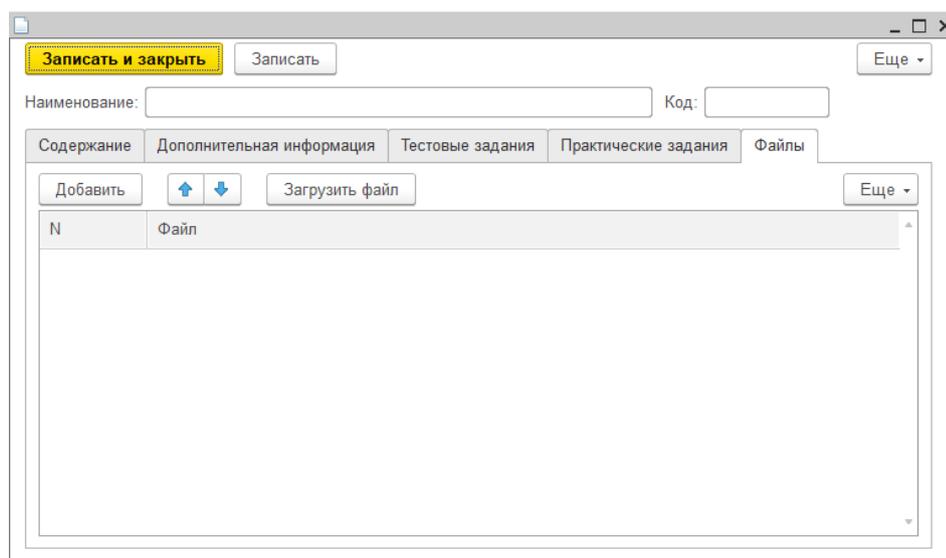
Вид формы для прав «Наставник» представлен на рис. 2.8.



The screenshot shows a web form window with a title bar. At the top left, there are two buttons: 'Записать и закрыть' (highlighted in yellow) and 'Записать'. At the top right, there is a dropdown menu labeled 'Еще'. Below these are two input fields: 'Наименование:' and 'Код:'. A horizontal tab bar contains five tabs: 'Содержание' (selected), 'Дополнительная информация', 'Тестовые задания', 'Практические задания', and 'Файлы'. The main area of the form is a large empty rectangular box.

Рисунок 2.8 – Вид формы справочника «Материалы занятий» для прав «Наставник»

Для стажеров и наставников на странице файлы доступна возможность выгрузки файлов на компьютер. Команда «Загрузить файлы» (рис. 2.9) доступна только для прав «Наставник».



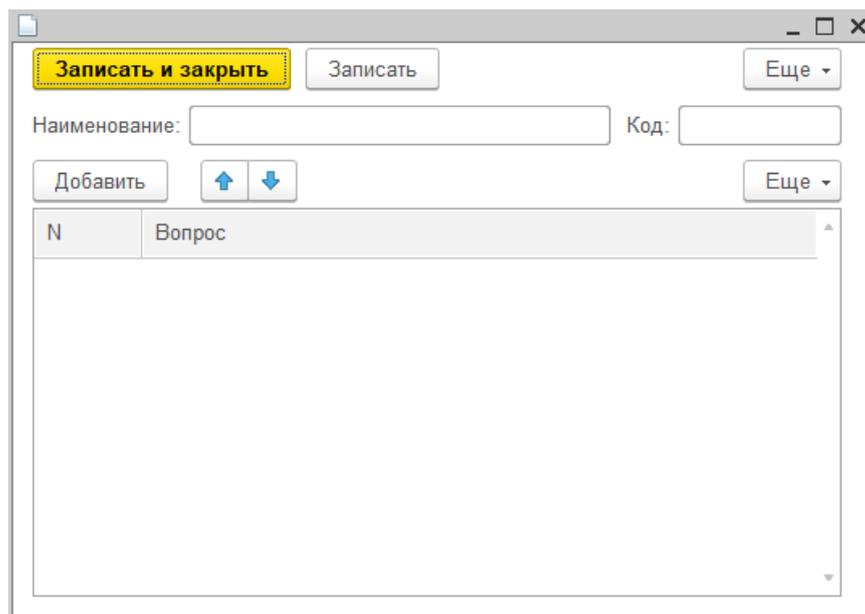
The screenshot shows the same web form window as in Figure 2.8, but with the 'Файлы' tab selected. Above the file list, there are buttons for 'Добавить', 'Загрузить файл', and a dropdown menu labeled 'Еще'. The file list itself has two columns: 'N' and 'Файл'. The list is currently empty.

Рисунок 2.9 – Вид формы справочника «Материалы занятий» для прав «Наставник», страница «Файлы»

Реализация программного кода модуля формы представлена в Приложение Г.

Наставник имеет возможность редактирования каждого элемента формы.

Справочники «Материалы тестовых заданий» (рис 2.10), «Вопросы тестовых заданий» (рис 2.11) и «Материалы практических заданий» (рис 2.12) будут иметь один вид, так как доступ к ним имеют только наставники.



The screenshot shows a software window titled «Материалы тестовых заданий». At the top, there are three buttons: «Записать и закрыть» (highlighted in yellow), «Записать», and «Еще ▾». Below these are two input fields: «Наименование:» and «Код:». Underneath are a «Добавить» button, two arrow buttons (up and down), and another «Еще ▾» button. The main area of the window contains a table with two columns: «N» and «Вопрос». The table is currently empty.

Рисунок 2.10 – Вид формы справочника «Материалы тестовых заданий» для прав «Наставник»



The screenshot shows a software window titled «Вопросы тестовых заданий». At the top, there are three buttons: «Записать и закрыть» (highlighted in yellow), «Записать», and «Еще ▾». Below these are two input fields: «Наименование:» and «Код:». Underneath is a large text area labeled «Текст вопроса:». At the bottom, there is an input field labeled «Ответ:».

Рисунок 2.11 – Вид формы справочника «Вопросы тестовых заданий» для прав «Наставник»

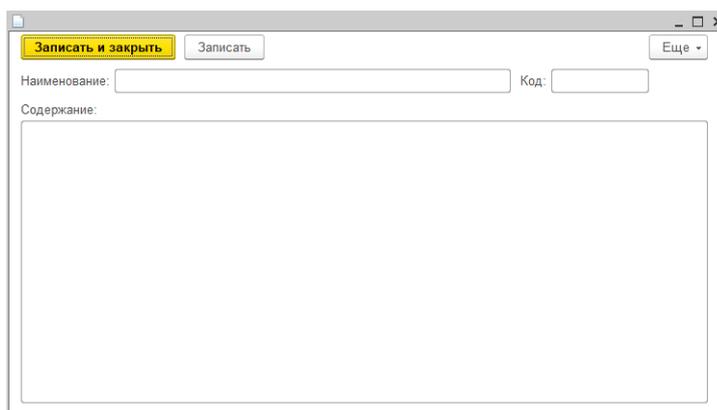


Рисунок 2.12 – Вид формы справочника «Материалы практических заданий» для прав «Наставник»

Обратим внимание, что при создании формы для задачи «Тестовое задание», потребуется добавление дополнительного функционала заполнения. При нажатии команды из формы справочника «Материалы занятий» «Выполнить тест» производится перезаполнение табличной части вопросы, если задача является новой, то есть имеет пустую ссылку на объект. Подобный алгоритм заполнения используется и в событии «при изменении» для реквизита «тестовое задание», для случаев ручного создания документов администраторами. Самостоятельное заполнение табличной части «Вопросы» исключено. Форма задачи «Тестовое задание» представлена на рис. 2.13 и рис. 2.14.

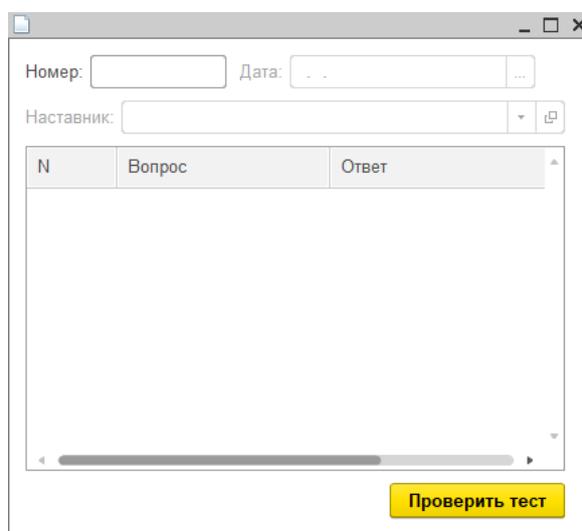


Рисунок 2.13 – Вид формы задачи «Тестовое задание» для прав «Стажер»

Рисунок 2.14 – Вид формы задачи «Тестовое задание»

Нажатие кнопки «Проверить тест», проводит проверку заполненных ответов с ответами, содержащимися в вопросах и помечает тест как выполненный. Реализация программного кода в модуле формы «Тестовое задание» представлено в Приложение Д.

Последней формой для разработки будет являться форма задачи «Практическое задание», представленная на рис. 2.15 и рис 2.16.

Рисунок 2.15 – Вид формы задачи «Практическое задание» для прав «Стажер»

Поле «комментарий наставника» появляется только в том случае, если поле заполнено. Команда «Отправить на проверку» записывает и помечает задачу как «требующуюся к проверке» в таком случае задача попадает в список наставника.

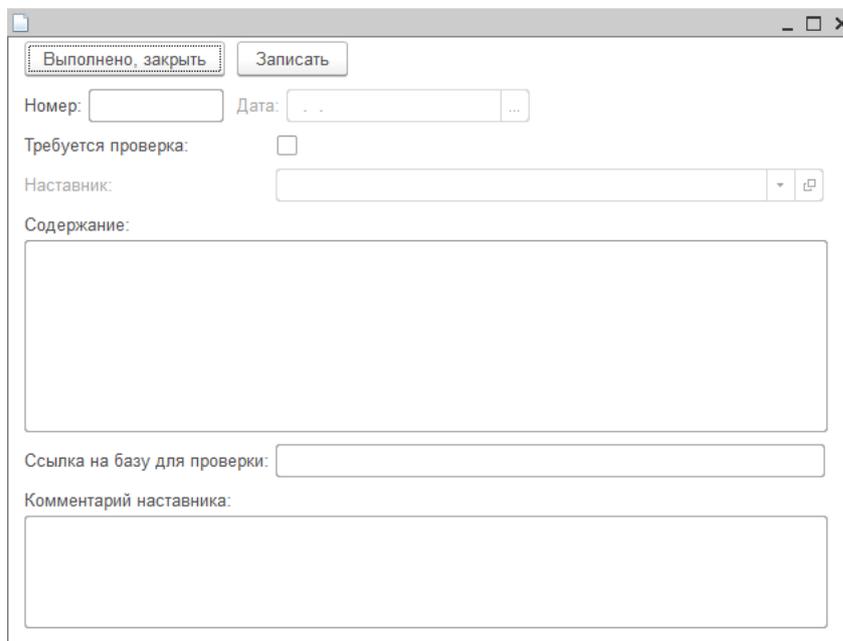
The image shows a screenshot of a web browser window displaying a form for task management. The window has a title bar with standard minimize, maximize, and close buttons. The form itself is contained within a rectangular frame. At the top left of the form is a button labeled "Выполнено, закрыть" (Completed, close) and at the top right is a button labeled "Записать" (Save). Below these are two input fields: "Номер:" (Number) and "Дата:" (Date). The "Дата:" field includes a calendar icon. Underneath is a checkbox labeled "Требуется проверка:" (Requires review). Below the checkbox is a dropdown menu labeled "Наставник:" (Mentor) with a small square icon to its right. A large, empty text area labeled "Содержание:" (Content) follows. Below this is a single-line input field labeled "Ссылка на базу для проверки:" (Link to the database for review). At the bottom is another large, empty text area labeled "Комментарий наставника:" (Mentor's comment).

Рисунок 2.16 – Вид формы задачи «Практическое задание» для прав «Наставник»

Для регистра сведений «Доступ к занятиям» разработка формы не требуется. Платформа «1С:Предприятие» будет формировать типовую форму записи.

Следующим этапом организации пользовательского интерфейса будет создание обработки «РМК».

Обработки предназначены для выполнения различных действий над информацией. Например, с их помощью можно выполнять удаление из системы устаревших данных, импорт информации из других систем и многое другое.

Обработка «РМК» (рабочее место кандидата) будет содержать две формы:

- форма кандидата (рис 2.17), данная форма отображает список доступных заданий, список выполненных тестовых заданий, а также список выполненных и находящихся на проверке практических заданий;
- форма наставника (рис 2.18) содержит в себе отчет по успеваемости кандидатов, список практических заданий, требующих проверки, возможность выдавать доступ к заданиям для кандидатов и доступ к справочникам с материалами для занятий и заданий.

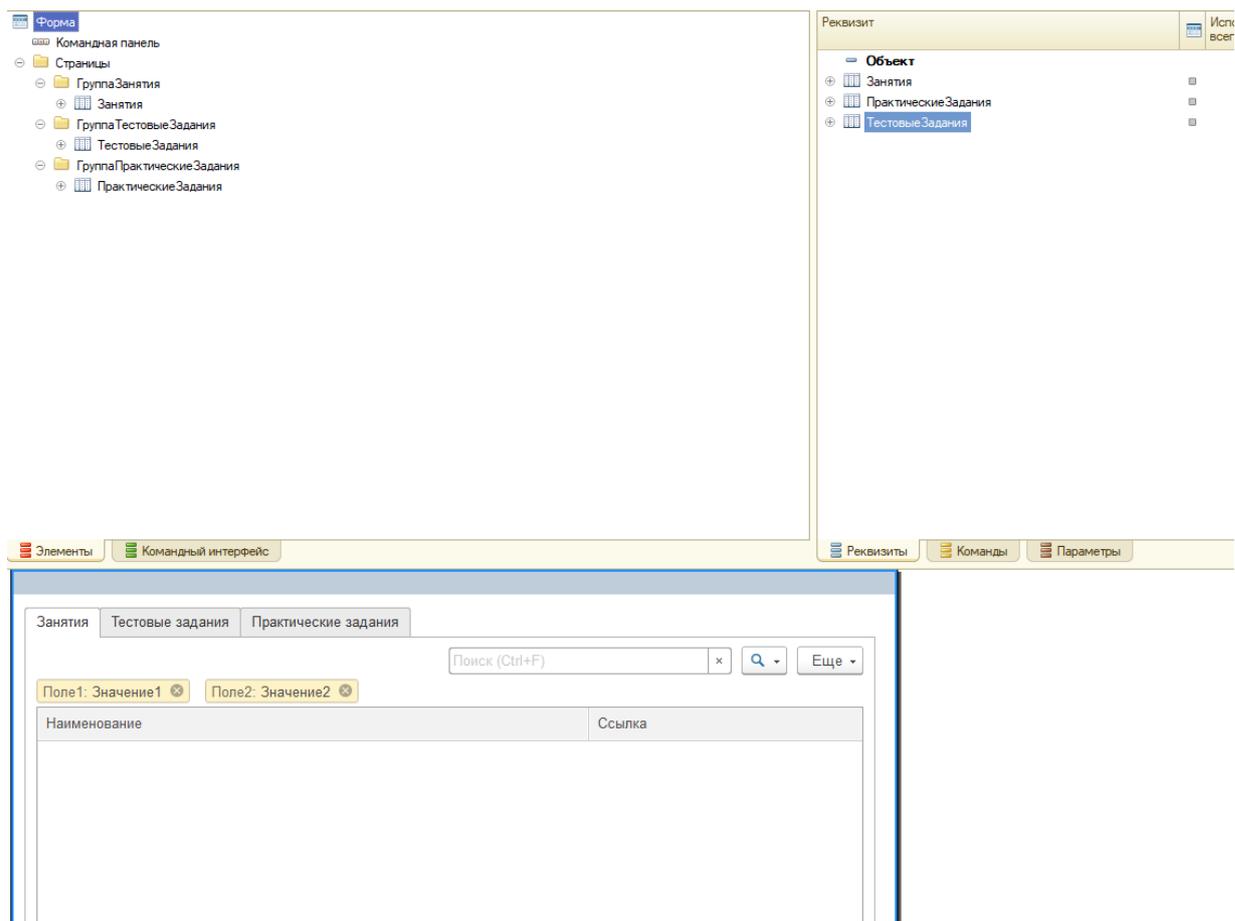


Рисунок 2.17 – Отображение формы кандидата в конфигураторе

Каждая из таблиц значений «Занятия» (справочник), «Тестовые задания» (задача), «Практические занятия» (задача) имеет в себе проверку на установленный в регистре сведений доступ к занятиям. Такой подход исключает отображение материала, недоступного для текущего пользователя.

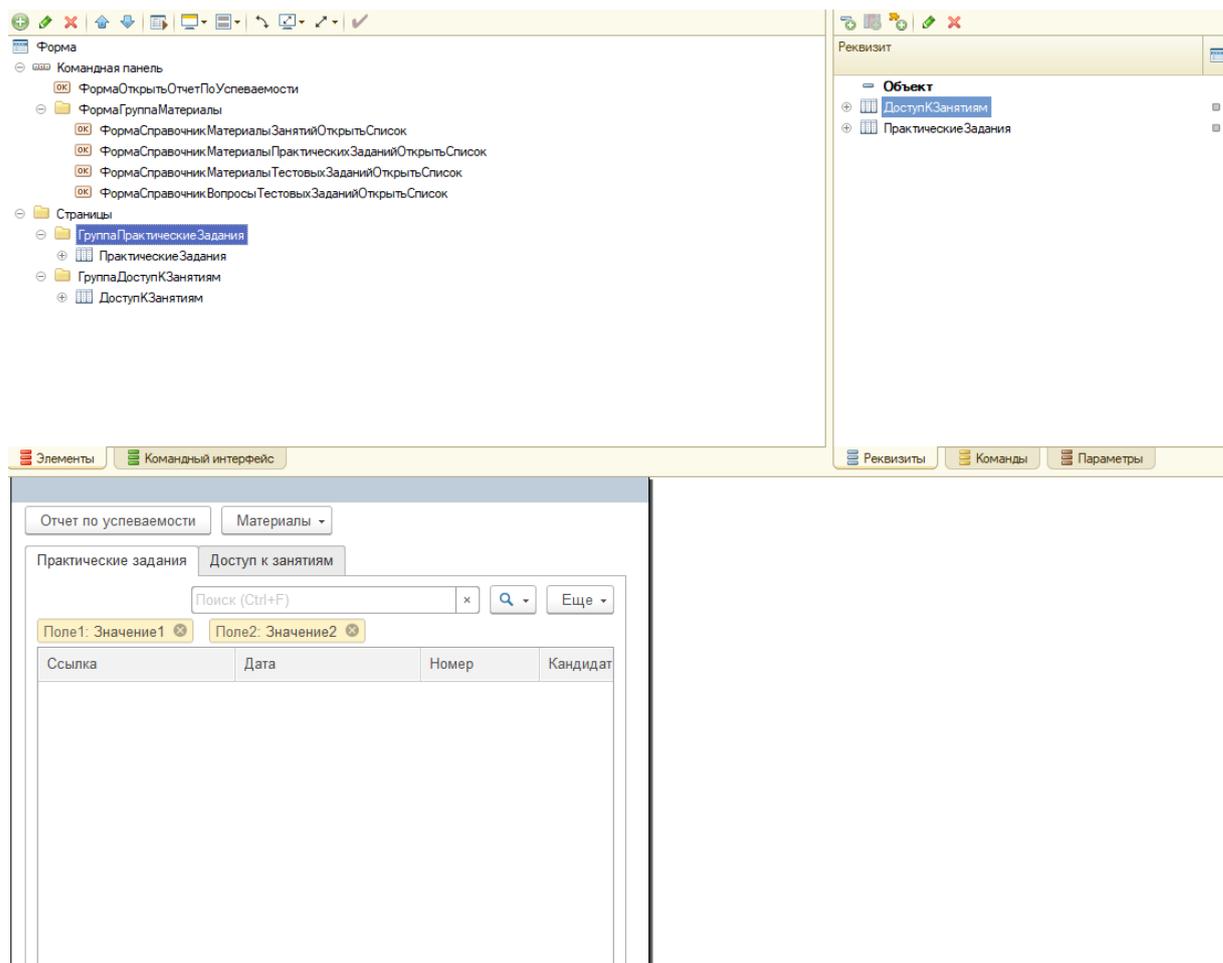


Рисунок 2.18 – Отображение формы наставника в конфигураторе

На форме наставника присутствуют все указанные требования:

- динамический список «Практические занятия» отражают занятия требующие проверки для текущего пользователя;
- динамический список «Доступ к занятиям» позволяет просмотреть выделенные пользователям доступы и добавить новые;
- команда формы «Отчет по успеваемости» открывает форму отчета;
- в подменю «Материалы» находятся все доступные наставнику материалы для редактирования (справочник «Материалы занятия», справочник «Материалы тестовых заданий» и т.д.).

Обе формы требуется добавить в рабочую область начальной страницы и устанавливается доступность по правам для возможности отображения при запуске программы.

## 2.3 Результаты тестирования обучающей программы

Тестирование программ основной метод, способный обеспечивать и удостоверить необходимое качество компонентов и комплексов программ, выявлять и устранять в них дефекты и ошибки при проектировании и производстве.

Технологически тестирование имеет две цели:

- первая состоит в поиске, обнаружении и устранении дефектов и ошибок в программах и в применении покрытия тестами требований к программному комплексу;
- вторая определяется возможностью статистической оценки того, что при тестировании не проявится дефект или ошибка компонента или комплекса программ.

Обе интерпретации этих целей одинаково важны для тестирования и достижения высокого качества программ. Тестирование для поиска, идентификации и устранения дефектов подразумевает успешность процедуры тестирования, если дефект найден. Это отличается от подхода в тестировании, когда тесты исполняются для демонстрации того, что компонент или программный комплекс полностью удовлетворяет предъявляемым требованиям и, соответственно, тест считается успешным, если не найдено дефектов. Тестирование программ может использоваться для демонстрации наличия дефектов, но никогда не гарантирует их отсутствие. Основная причина этого в том, что полное, всеобъемлющее тестирование недостижимо для реальных сложных программных комплексов. Когда оцениваются и применяются методы тестирования, надо четко определять, что подразумевается под их эффективностью, желательно, в количественных величинах. Только обладая такого рода данными можно говорить о корректности сравнения и оценки эффективности разных методов тестирования.

Основная цель тестирования базы состоит в том, чтобы обнаруживать, регистрировать и устранять дефекты и ошибки, которые внесены во время последовательной разработки и реализации требований к его функциям и характеристикам.

Перед проведением тестирования разработки составим план тестирования по шаблону, представленному в Приложении Б.

Разработанный план отражен на рис. 2.19.

<b>План тестирования</b>		
Программист	Колчинцева К.В.	
Тестировщик	Колчинцева К.В.	
<b>1. Что нужно сделать для включения механизма в работу</b>		
+	Настройки, необходимые для запуска	
	<ol style="list-style-type: none"> <li>1. Создать тестовую базу на сервере;</li> <li>2. Загрузить конфигурацию программы обучения;</li> <li>3. Добавить путь к базе в общий список баз организации.</li> </ol>	
<b>2. Настройки внедряемого механизма</b>		
	Настройки, необходимые для запуска	
	<ol style="list-style-type: none"> <li>1. Провести заполнение программы обучения (добавить материалы занятий, тестов и практических заданий);</li> <li>2. Создать пользователей для тестирования и назначить соответствующие права.</li> </ol>	
<b>3. Порядок тестирования</b>		
№	Что нужно сделать	Реакция системы
1	Тестирование справочников	Справочники должны отображать формы согласно заданным ролям, все команды на формах должны выполняться, при открытии форм должны заполняться данные.
2	Тестирование доступа	В зависимости от наличия записей в регистре сведений «Доступ к занятиям» должен корректно отображаться список заданий
3	Тестирование задач	Создаем задачи из справочников, под администратором.
4	Тестирование отчета	Отчет должен отображать актуальную информацию по журналам заданий.

Рисунок 2.19 – План тестирования обучающей программы

При выполнении тестирования будет использоваться модульное тестирование, описанное В.П. Котляровым в книге «Основы тестирования программного обеспечения». [7]

План тестирования позволит отслеживать выполненные шаги при тестировании, постепенно выявляя ошибки.

Тестирование справочников и задач выявили ошибки интерфейса пользователя:

- все многострочные поля (содержание, дополнительная информация и т.д.) должны иметь доступность, но иметь флаг «Только просмотр»;
- задачи «Тестовое задание» и «Практическое задание» должны иметь команду «записать», для возможности поэтапного выполнения заданий;

После окончания тестирования формы объектов доработаны и дефекты устранены.

## **Выводы по главе 2**

Разработка баз данных трудоемкий и долгий процесс. Система «1С:Предприятие» предоставляет сторонним разработчикам возможность использовать в своих решениях типовой функционал, используя конфигурацию «1С:Библиотека стандартных подсистем». Однако специфический функционал, такой как возможность обучения стажеров, отсутствует в типовых механизмах.

При разработке обучающей программы было установлено:

- залогом качества выполненной работы является разработка концептуальной модели базы данных, так как структура хранения данных является основой для правильного ведения учета деятельности;
- удобство использования программы так же является немаловажной частью разработки;

- выполнение тестирования должно присутствовать при разработке любого программного обеспечения, так как оно помогает увидеть полную картину и доработать неучтенные аспекты при разработке.

## Глава 3 Применение обучающей программы и анализ эффективности

### 3.1 Заполнение теоретического материала, составление тестовых и практических заданий

Новая обучающая программа в организации ООО «ТриоТ» рассчитана на четыре недели. За это время кандидат должен получить базовые знания в работе с продуктами «1С», выполнить тестовые и практические задания к каждому занятию. Материалы для обучения предоставляются организацией. Каждая тема обязательно содержит минимум одно тестовое и практическое занятие, закрепляющее полученную информацию. Полное внесение материалов занятий в базу данных заняло около десяти часов, а создание тестовых и практических заданий пять.

Программа обучения составлялась по книге «1С: Предприятие 8.3. Практическое пособие разработчика», написанная М. Г. Радченко. [10]

План программы обучения представлен в таблице 3.1.

Таблица 3.1 – План программы обучения

№	Тема	Тестовое задание выполнено	Практическое задание выполнено	Дата выполнения
<b>Модуль 1</b>				
1	Знакомство, создание информационной базы			
2	Подсистемы			
3	Справочники			
4	Документы			
5	Перечисления			
6	Регистры накопления			
7	Оборотные регистры накопления			
8	Редактирование движений в форме документа			
<b>Модуль 2</b>				
9	Формы и общие модули			
10	Периодические регистры сведений			

№	Тема	Тестовое задание выполнено	Практическое задание выполнено	Дата выполнения
11	План видов характеристик			
12	План видов расчета, регистр расчета			
13	Использование регистра расчета			
14	Начальная страница и настройка командного интерфейса			
15	Бухгалтерский учет			
16	Поиск в базе данных			
<b>Модуль 3</b>				
17	Выполнение заданий по расписанию			
18	Отчеты			
19	Простой отчет			
20	Макеты			
21	Редактирование макетов			
22	Обмен данными			
23	Функциональные опции			
<b>Модуль 4</b>				
24	Организация подборов, особенности разработки в режиме без использования модальности и ввод данных на основании			
25	Приемы разработки форм			
26	Приемы редактирования форм			
27	Краткий справочник разработчика			
28	Итоговое задание			

Данный план ведется наставником для каждого кандидата в отдельности под его руководством. Результат выполнения заданий отмечается в колонках «Практическое задание выполнено» и «Тестовое задание выполнено». В графе «Дата выполнения» отмечается время окончания занятия.

По прохождению полного курса каждый стажер должен будет выполнить итоговое задание, содержащее в себе несколько задач на пройденные темы. В обучающей программе оно создается как задача «Практическое занятие» прикрепленное к элементу справочника «Материалы занятий» под наименованием «Итоговое занятие».

По завершению итогового теста наставник выводит отчет по успеваемости и заполняет план программы обучения.

Примеры практических заданий отражены на рис. 3.1 и рис. 3.2.

#### **Оперативный учет**

Компания занимается оптовой торговлей. Поступление товара отражается документом «Приходная накладная», продажа – «Расходная накладная». Помимо продажи товара, могут оказываться дополнительные услуги, например по доставке. И услуги и товары указываются в одной табличной части.

Учет товаров ведется в разрезе складов. Поступление и продажа осуществляется с указанием склада (в шапке документа).

При проведении документа «Расходная накладная» необходимо производить списание товара со склада. В том случае, когда товара не хватает, документ проводиться не должен.

Списание себестоимости товаров должно быть организовано по партиям, в зависимости от метода списания (FIFO или LIFO), принятого в учетной политике. Значение учетной политики меняется не чаще одного раза в год. При проведении документа необходимо использовать метод, актуальный на момент проведения.

Для расчета себестоимости при списании товара необходимо учитывать только момент поступления товара в компанию, вне зависимости от того, на какой склад он пришел. Предположим, для метода списания FIFO первое поступление портсигара произошло на склад «Основной» документом «Приходная накладная №1», а потом на склад «Транзитный» документом «Приходная накладная №2». В этом случае при продаже товара со склада «Транзитный» в первую очередь должна быть списана себестоимость портсигара по документу «Приходная накладная №1», так как она пришла раньше.

Рисунок 3.1 – Пример практических заданий

#### Бухгалтерский учет

Необходимо создать документ «Операция», с помощью которого пользователь должен иметь возможность ввести проводки с произвольной корреспонденцией счетов. При решении задачи следует учесть возможность наличия проводок, сформированных с помощью данного документа.

Компания занимается торговлей продуктами питания. Учет товаров ведется в разрезе сроков годности. Под сроком годности принимается календарная дата, до которой товар годен к употреблению. На один и тот же товар с разными сроками годности при его поступлении может указываться разная цена. Например, на товар со сроком годности 10 января 2010 цена может быть ниже, чем на товар со сроком годности 30 января 2010. Товар с одинаковым сроком годности может поступать разными документами и по разной цене. Возможна ситуация когда в одном документе один и тот же товар поступает с разными сроками годности (и по разной цене). Учет товаров в разрезе складов не ведется.

Документ «Приходная накладная» реализует следующую проводку:

*Дт «Товары» - Кт «Поставщики» на количество и сумму закупаемого товара.*

Продажа товара регистрируется документом «Расходная накладная». При продаже срок годности не указывается. В первую очередь списывается товар с наименьшим календарным сроком годности. Себестоимость определяется как средняя по товару по всем срокам годности. Т.е. например если 1 пачка йогурта со сроком годности 10.01.2010 поступила по цене 90 рублей и еще 1 пачка того же йогурта, но со сроком годности 30.01.2010 поступила по цене 110 рублей, то при списании себестоимость одной пачки данного йогурта равна 100 рублей.

Документ «Расходная накладная» реализует следующие проводки:

*Дт «Прибыли и убытки» - Кт «Товары» на количество и сумму себестоимости*

*Дт «Покупатели» - Кт «Прибыли и убытки» на сумму в продажных ценах.*

При проведении документа анализируется наличие этого товара в организации. Если товара не достаточно документ не проводится.

### Рисунок 3.2 – Пример практических заданий

Еще больше примеров заданий представлены фирмой «1С» в своем сборнике задач по подготовке к экзамену «1С:Специалист». [8]

После заполнения теоретического материала обучающая программа готова к использованию стажерами.

### 3.2 Итоги обучения стажеров-программистов и внесение изменений в обучающую программу

В организацию были приняты три новых кандидата на должность «Программист». Их главной задачей – работа с обучающей программой в качестве пользователей. При работе в базе данных каждый кандидат выполнял требуемые от него действия:

- выполнение обучения, практических и тестовых заданий;
- фиксация критических ошибок и уведомление о них наставника;
- анализ функционала базы данных со стороны пользователя и формирования списка пожеланий к концу курса.

По завершению курса среди кандидатов был произведен опрос об использовании обучающей программы, результаты которого отражены в таблице 3.2.

Таблица 3.2 – Результаты опроса кандидатов об использовании программы

Вопрос	Кандидат 1	Кандидат 2	Кандидат 3
Удобна ли в использовании обучающая программа?	Да	Да	Да
Сложно ли освоиться в работе с программой?	Нет	Нет	Нет
Как много критических ошибок было обнаружено?	1	0	0
Как много ошибок интерфейса было обнаружено?	0	2	1
Какой функционал вы хотели бы добавить в обучающую программу?	Возможность текстовых выделений (курсив, жирность и т.д.)	Возможность отображения графического материала не через файлы.	Поиск по материалам занятий.

В процессе применения обучающей программы наставниками и стажерами были отмечены незначительные недостатки программы и сразу проведено их устранение.

1. Элементы справочника «Материалы занятий» должны группироваться по модулям. Для этого требуется сделать справочник «Материалы занятий» иерархическим. Иерархический справочник – это также многоуровневый справочник, или справочник с группами. Группа справочника в «1С» часто называется «родитель». Список элементов справочника в системе «1С:Предприятие» может быть многоуровневым.

2. В отчете «Отчет по успеваемости» нет возможности вывести дату;

3. Вместо ручного заполнения финального отчета по практике на основе данных из отчета по успеваемости требуется разработать новый отчет «Итоговые результаты».

Отдельно рассмотрим формирование отчета «Итоговые результаты». Макет отчета представлен на рис. 3.3.

	A	B	C	D	E
1	Кандидат				
2	Модуль				
3	Тема				
4	Задание				Дата выполнения
5	Иванов				
6	Модуль 1				
7	Знакомство, создание информационной базы				
8	Тестовое задание 1				10.05.2020
9	Практическое задание 1				10.05.2020
10	Подсистемы				
11	Тестовое задание 2				10.05.2020
12	Практическое задание 2				11.05.2020
13	Справочники				
14	Документы				
15	Перечисления				
16	Регистры накопления				
17	Оборотные регистры накопления				
18	Редактирование движений в форме документа				
19	Модуль 2				
20	Формы и общие модули				
21	Периодические регистры сведений				
22	План видов характеристик				
23	План видов расчета, регистр расчета				
24	Использование регистра расчета				
25	Начальная страница и настройка командного интерфейса				
26	Бухгалтерский учет				
27	Поиск в базе данных				
28					

Рисунок 3.3 – Макет отчета «Итоговые результаты»

В отчете «Итоговые результаты» должны отображаться все существующие задания согласно и группироваться по родителю (модулю), если существует выполнение задание оно так же должно отображаться в данном отчете.

Схема компоновки данных и вариант отчет для отчета «Итоговые результаты» представлены на рис. 3.4 и рис. 3.5.

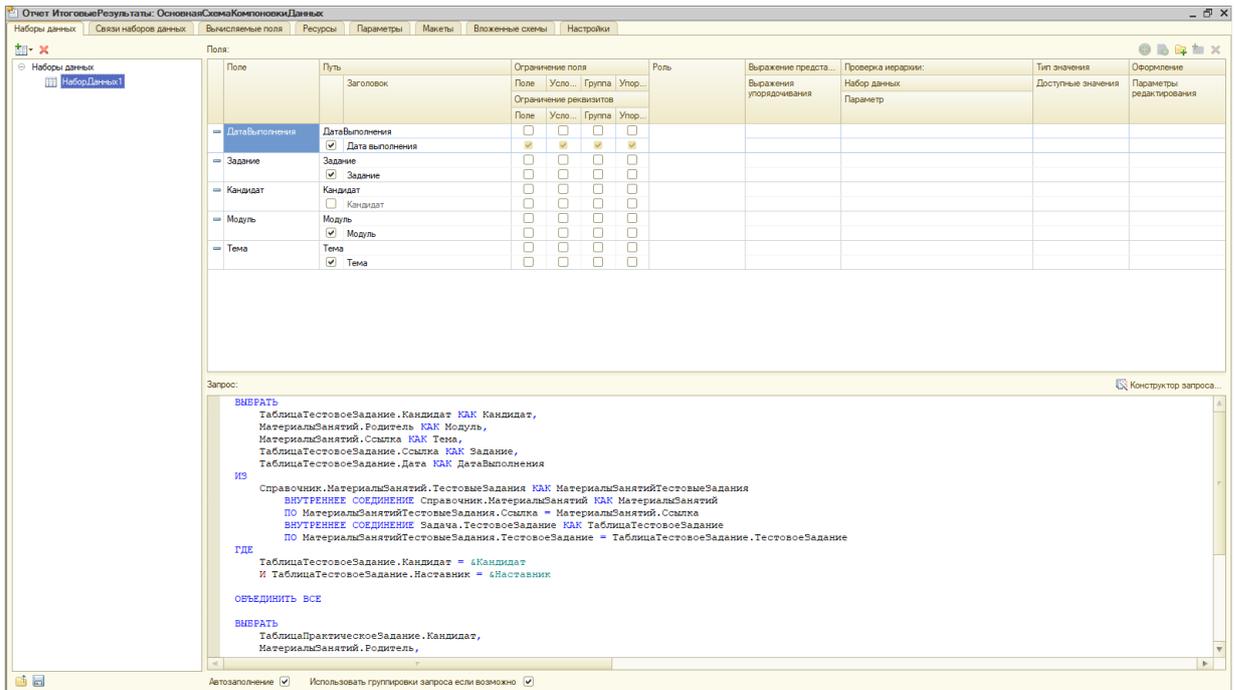


Рисунок 3.4 – Схема компоновки отчета «Итоговые результаты» в конфигураторе

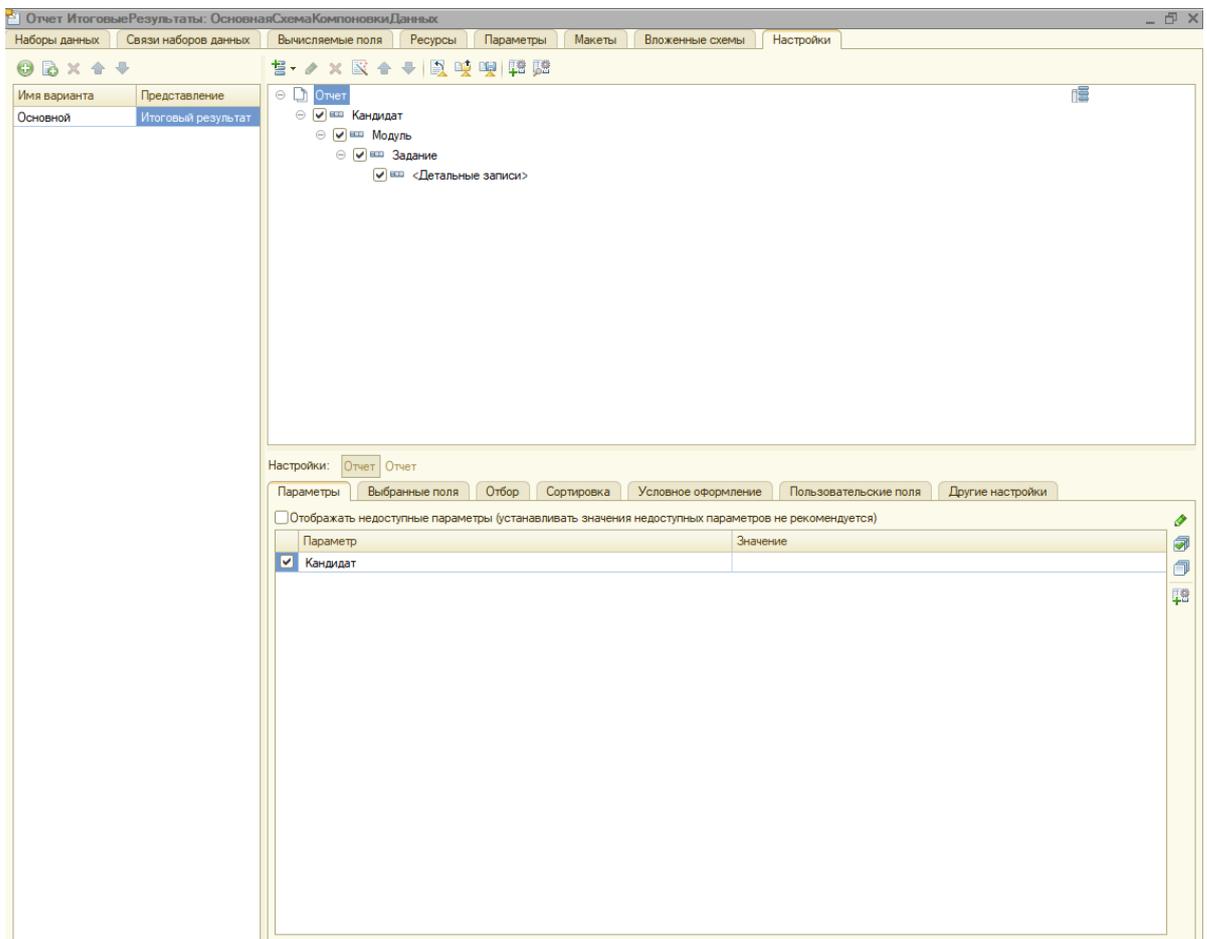


Рисунок 3.5 – Вариант отчета «Итоговые результаты» в конфигураторе

Отчет всегда формируется в разрезе текущего пользователя (наставника) с обязательным заполнением параметра «кандидат».

### 3.3 Анализ эффективности обучения стажеров по новой модели бизнес-процесса

Подведем итоги по проделанной работе. Организация предоставила данные временных затрат по проводимым обучением (таблица 3.3).

Таблица 3.3 – Временные затраты предыдущего обучения стажеров

Тема задания	Кандидат 1	Кандидат 2	Кандидат 3
Алгоритмы	30 ч.	28 ч.	31 ч.
Структура баз данных	85 ч.	92 ч.	83 ч.
Управляемы и обычные формы	23 ч.	25 ч.	25 ч.
Регистры	47 ч.	46 ч.	44 ч.
Партионный учет и себестоимость	44 ч.	44 ч.	43 ч.
Взаиморасчеты	39 ч.	42 ч.	39 ч.
Бухгалтерский учет	84 ч.	94 ч.	83 ч.
<b>Итого</b>	<b>342 ч.</b>	<b>361 ч.</b>	<b>338 ч.</b>

Для выполнения каждого задания кандидату требовалось самостоятельно находить и изучать материал в интернете, а также просматривать официальные источники фирмы «1С».

При анализе таблицы видим, что при восьмичасовом рабочем дне среднее количество дней на выполнение всех практических заданий составляет тридцать семь дней. Такие результаты превышают запланированные сроки обучения в четыре недели больше чем в половину.

Рассмотрим временные затраты обучения по новой модели бизнес-процесса (таблица 3.4).

Таблица 3.4 – Временные затраты обучения по новой модели бизнес-процесса

№	Тема	Кандидат 1	Кандидат 2	Кандидат 3
<b>Модуль 1</b>				
1	Знакомство, создание информационной базы	8 ч.	8 ч.	7 ч.

№	Тема	Кандидат 1	Кандидат 2	Кандидат 3
2	Подсистемы	5 ч.	4 ч.	3 ч.
3	Справочники	7 ч.	8 ч.	7 ч.
4	Документы	8 ч.	10 ч.	9 ч.
5	Перечисления	4 ч.	4 ч.	3 ч.
6	Регистры накопления	11 ч.	13 ч.	10 ч.
7	Оборотные регистры накопления	6 ч.	7 ч.	6 ч.
8	Редактирование движений в форме документа	7 ч.	7 ч.	8 ч.
Итого по модулю		56 ч.	61 ч.	52 ч.
<b>Модуль 2</b>				
9	Формы и общие модули	6 ч.	7 ч.	7 ч.
10	Периодические регистры сведений	5 ч.	5 ч.	4 ч.
11	План видов характеристик	5 ч.	5 ч.	4 ч.
12	План видов расчета, регистр расчета	9 ч.	11 ч.	9 ч.
13	Использование регистра расчета	10 ч.	12 ч.	10 ч.
14	Начальная страница и настройка командного интерфейса	3 ч.	3 ч.	3 ч.
15	Бухгалтерский учет	26 ч.	29 ч.	25 ч.
16	Поиск в базе данных	3 ч.	3 ч.	3 ч.
Итого по модулю		67 ч.	75 ч.	65 ч.
<b>Модуль 3</b>				
17	Выполнение заданий по расписанию	6 ч.	7 ч.	6 ч.
18	Отчеты	11 ч.	12 ч.	10 ч.
19	Простой отчет	4 ч.	4 ч.	3 ч.
20	Макеты	6 ч.	7 ч.	5 ч.
21	Редактирование макетов	5 ч.	6 ч.	5 ч.
22	Обмен данными	9 ч.	11 ч.	9 ч.
23	Функциональные опции	5 ч.	8 ч.	7 ч.
Итого по модулю		46 ч.	55 ч.	45 ч.
<b>Модуль 4</b>				
24	Организация подборов, особенности разработки в режиме без использования модальности и ввод данных на основании	5 ч.	6 ч.	4 ч.
25	Приемы разработки форм	5 ч.	5 ч.	4 ч.
26	Приемы редактирования форм	10 ч.	10 ч.	10 ч.
27	Краткий справочник разработчика	7 ч.	6 ч.	5 ч.
28	Итоговое задание	30 ч.	34 ч.	29 ч.
Итого по модулю		57 ч.	61 ч.	52 ч.
<b>Итого</b>		<b>226 ч.</b>	<b>242 ч.</b>	<b>214 ч.</b>

Проведем сравнение временных затрат для старой и новой модели бизнес-процесса обучения и подведем итоги:

- временные затраты сократились до минимума (в среднем двадцать восемь дней на обучение);
- время прохождения курса равно запланированному времени;
- результаты новой обучающей программой легче контролировать за счет более детализированной программы обучения;

Разработанная обучающая программа имеет еще несколько плюсов:

- качество просматриваемых кандидатами материалов контролируется наставниками и организацией;
- прохождение тестовых заданий позволяет кандидатам проверить собственные знания без участия наставника;
- контроль со стороны наставника за прохождением курса позволяет вовремя среагировать на трудности у кандидата и оказать помощь в изучении материала или выполнении заданий.

### **Выводы по главе 3**

Подведем итоги практического применения обучающей программы:

- при вводе в эксплуатацию были обнаружены и устранены нюансы, неучтенные при разработке и проектировании обучающей программы;
- время выполнения обучения значительно сократилось и выполняется кандидатами в запланированные сроки;
- контроль со стороны наставника позволяет отслеживать проблемы у стажеров и оперативно их решать.

Сокращение временных затрат приводит к снижению стоимостных затрат. Стоимость и эффективность работы является важным показателем результата, как описывает L. Campbell в книге «Database Reliability Engineering (2017)». [20]

Разработка обучающей программы оправдана и в дальнейшем может производиться дополнительная доработка для данной программы.

## Заключение

В процессе выпускной квалификационной работы «Автоматизация обучения стажеров-программистов 1С:Предприятие» было выполнено несколько этапов работы.

1. Проведена характеристика предприятия и разработаны модели бизнес-процесса «обучений» (модель «AS-IS» и модель «TO-BE»). Выполненное исследование позволило определить проблемы в модели бизнес-процесса обучения «как-есть» и сформировать представление процесса «как должно быть».

2. Разработано техническое задание по разработке обучающей программы, которое позволило установить требования к разрабатываемой базе данных и определить план работы.

3. Составлена концептуальная модель базы данных и разработана обучающая программа для стажеров на платформе «1С:Предприятия».

4. Проведено предварительное тестирование и устранены недочеты, неучтенные при разработке технического задания и проектировании концептуальной модели. Выявленные ошибки не являются критичными и касаются в основном пользовательского интерфейса и удобства пользования программой.

5. Программа введена в эксплуатацию и проведен сравнительный анализ результатов обучения, итогом которого было определен положительный эффект автоматизации процесса обучения.

Разработанная программа оказала на выполнение бизнес-процесса «Обучение» положительный эффект:

- сократились временные затраты на обучение;
- занесенный в обучающую программу теоретический и практический материал помогает отслеживать прогресс прохождения курса кандидатами, им больше не требуется выполнять самостоятельный поиск информации;

- вся информация по обучению хранится в одном месте.

Основная цель выпускной квалификационной работы, автоматизировать процесс обучения для кандидатов на должность «программист» и сократить временных затрат на проведение обучения, выполнена.

По окончании прохождения курса кандидаты получили, правильно сформированный, комплекс знаний в сфере «1С» и прошли сертификации «1С:Профессионал».

## Список используемой литературы

1. Елиферов, В.Г. Бизнес-процессы [Электронный ресурс]: регламентация и управление: учебник / В.Г. Елиферов, В.В. Репин. – М.: ИНФРА-М, 2018. – 319 с.;
2. Татарникова, Т. М. Системы управления базами данных: учебное пособие / Т. М. Татарникова. — Санкт-Петербург: Российский государственный гидрометеорологический университет, 2004. — 88 с. — ISBN 2227-8397. — Текст: электронный;
3. Джестон, Дж. Управление бизнес-процессами [Текст]: практ. руководство по успешной реализации проектов / Джестон Дж., Н. Йохан; пер. с англ. [В. Агапова; науч. ред. В. Тренев]. – М.: Альпина Паблишер, 2016. – 648 с.;
4. «1С» ИТС, <https://its.1c.ru/>
5. Блинов, А.О. Реинжиниринг бизнес-процессов [Электронный ресурс]: учеб. пособие / А.О. Блинов [и др.]; под ред. А.О. Блинова. – М.: ЮНИТИ-ДАНА, 2015. – 343 с.
6. Варзунов, А.В. Анализ и управление бизнес-процессами / А.В. Варзунов, Е.К. Торосян, Л.П. Сажнева;
7. Котляров В.П., Основы тестирования программного обеспечения / В.П. Котляров;
8. Фирма «1С», Сборник задач для подготовки к экзамену «1С:Специалист» по конфигурированию и внедрению торговых решений в прикладных решениях «1С:Предприятие 8» / Фирма «1С»;
9. Габец, А.П. 1С: Предприятие 8.2. Простые примеры разработки / А.П. Габец, Д.И. Гончаров;
10. Радченко, М. Г. 1С: Предприятие 8.3. Практическое пособие разработчика / М. Г. Радченко, Е. Ю. Хрусталева;
11. Привалов И. М., Основы аппаратного и программного обеспечения / И. М. Привалов;

12. Баканов, А. С. Эргономика пользовательского интерфейса. От проектирования к моделированию человеко-компьютерного взаимодействия / А. С. Баканов, А. А. Обознов. — Москва : Институт психологии РАН, 2011. — 176 с. — ISBN 978-5-9270-0191-0. — Текст;
13. Грекул, В.И. Управление внедрением информационных систем [Текст]: учебник / В.И. Грекул, Г.Н. Денищенко, Н.Л. Коровкина. — М.: ИНТУИТ, 2017; Саратов: Вузовское образование, 2017. — 224 с.;
14. Фирма «1С», 1С:Предприятие 8.0. Описание встроенного языка / Фирма «1С»;
15. Ажеронок, В.А. Разработка управляемого интерфейса / В.А. Ажеронок, А. В. Островерх, М. Г Радченко, Е. Ю. Хрусталева;
16. Coronel, C. Database Systems: Design, Implementation, & Management (2019) / C. Coronel, S. Morris;
17. Elmasri, R. Fundamentals of Database Systems / R. Elmasri, Shamkant B. Navathe;
18. Sharma, N. Database fundamentals (2010) / N. Sharma, L. Perniu, Raul F. Chong, A. Iyer, C. Nandan, Adi-Cristina Mitea, M. Nonvinkere, M. Danubianu;
19. Hernandez, M. J. Database Design for Mere Mortals: A Hands-On Guide to Relational Database Design (2013) / M. J. Hernandez;
20. Campbell, L. Database Reliability Engineering (2017) / L. Campbell, C. Majors.

## Приложение А

### Техническое задание по разработке обучающей программы

Требования к обучающей программе:

7. Обучающая программа должна работать на платформе «1С: предприятие 8.3» (8.3.15.18.30);
8. Обучающая программа должна поддерживать типовой функционал: варианты отчетов, длительные операции, дополнительные отчеты и обработки, журнал регистрации, пользователи, резервное копирование, удаление помеченных объектов;
9. Обучающая программа должна поддерживать возможность работы нескольких пользователей одновременно;
10. Обучающая программа должна иметь возможность добавления прав «Наставник»: наставник имеет полный доступ к базе данных и способен давать доступ к курсам для стажеров;
11. Обучающая программа должна иметь возможность добавления прав «Стажер»: минимальные права, позволяющие читать материалы для обучения и практические задания, прохождение тестов;
12. Обучающая программа должна поддерживать возможность хранения выгрузок конфигураций баз данных для проверки практических заданий;
13. Обучающая программа должна давать функционал: создание/редактирования/чтения материалов для обучения, создание/редактирования/выполнение тестовых заданий, создание/редактирования/чтения практических заданий, предоставление доступа к материалам/тестам/заданиям, выставление оценок выполнения практических и тестовых заданий.

Таблица 1. – План работы

Этап	Плановая дата выполнения	Фактическая дата выполнения
Разработать концептуальную модель базы данных обучения	06.05.20	
Разработка обучающей программы	13.05.20	

Этап	Плановая дата выполнения	Фактическая дата выполнения
<ul style="list-style-type: none"> <li>• разработать структуру базы данных;</li> <li>• разработать пользовательский интерфейс;</li> <li>• разработать отчет для проверки успеваемости кандидатов</li> </ul>		
Провести предварительное тестирования и доработку обучающей программы	20.05.20	
Применение обучающей программы на практике	02.06.20	

**Приложение Б**  
**Шаблон плана тестирования**  
**План тестирования**

ист Программ  
ик Тестировщ

**1. Что нужно сделать для включения механизма в работу**

Внешние подключаемые отчеты и обработки, доступность пользователю, подсистема

Настройки, необходимые для запуска

Обработки, которые необходимо выполнить перед запуском в работу

Другая информация, необходимая для запуска

**2. Настройки внедряемого механизма**

Настройки, необходимые для запуска

**3. Порядок тестирования**

	Что нужно сделать	Реакция системы



## Приложение В

### Программный код проверки доступности ролей «Стажер» и «Наставник»

Функция ПроверитьРольНаставник() экспорт

Если РольДоступна("Наставник") тогда

Возврат истина;

Иначе

Возврат Ложь;

КонецЕсли;

КонецФункции

Функция ПроверитьРольСтажер() экспорт

Если РольДоступна("Стажер") тогда

Возврат истина;

Иначе

Возврат Ложь;

КонецЕсли;

КонецФункции

## Приложение Г

### Реализация модуля формы справочника «Материалы занятий»

#Область СобытияФормы

&НаСервере

Процедура ПриСозданииНаСервере(Отказ, СтандартнаяОбработка)

Если РМКВызовСервера.ПроверитьРольСтажер() Тогда

Элементы.Наименование.Доступность

= Ложь;

Элементы.Содержание.Доступность

= Ложь;

Элементы.ДополнительнаяИнформация.Доступность

= Ложь;

Иначе

Элементы.ФормаВыполнитьТест.Видимость = Ложь;

Элементы.ФормаВыполнитьПрактическоеЗадание.Видимость = Ложь;

КонецЕсли;

КонецПроцедуры

#КонецОбласти

#Область КомандыФормы

&НаКлиенте

Процедура ВыполнитьТест(Команда)

Если Элементы.ТестовыеЗадания.ТекущиеДанные =

Неопределено Тогда

Сообщить("Выберете тестовое задание.");

Возврат;

КонецЕсли;

ТестовоеЗадание =

ПолучитьЗадачуТестовоеЗадание(Элементы.ТестовыеЗадания.ТекущиеДанные.ТестовоеЗадание);

Если ТестовоеЗадание = Неопределено Тогда

ОткрытьФорму("Задача.ТестовоеЗадание.Форма.ФормаЗадачи", Новый Структура("Занятие, ТестовоеЗадание", Объект.Ссылка, Элементы.ТестовыеЗадания.ТекущиеДанные.ТестовоеЗадание));

Иначе

ОткрытьФорму("Задача.ТестовоеЗадание.Форма.ФормаЗадачи", Новый Структура("Ключ", ТестовоеЗадание));

КонецЕсли;

КонецПроцедуры

&НаСервере

Функция ПолучитьЗадачуТестовоеЗадание(ТестовоеЗадание)

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| ТаблицаТестовоеЗадание.Ссылка КАК Ссылка

|ИЗ

| Задача.ТестовоеЗадание КАК

ТаблицаТестовоеЗадание

|ГДЕ

```
| ТаблицаТестовоеЗадание.Кандидат = &Кандидат  
| И ТаблицаТестовоеЗадание.ТестовоеЗадание =  
&ТестовоеЗадание";
```

```
Запрос.УстановитьПараметр("Кандидат",  
ПараметрыСеанса.ТекущийПользователь);
```

```
Запрос.УстановитьПараметр("ТестовоеЗадание",  
ТестовоеЗадание);
```

```
ВыборкаДетальныеЗаписи = Запрос.Выполнить().Выбрать();
```

```
Если ВыборкаДетальныеЗаписи.Следующий() Тогда
```

```
    Возврат ВыборкаДетальныеЗаписи.Ссылка;
```

```
КонецЕсли;
```

```
Возврат Неопределено;
```

```
КонецФункции
```

```
&НаКлиенте
```

```
Процедура ВыполнитьПрактическоеЗадание(Команда)
```

```
    Если Элементы.ПрактическиеЗадания.ТекущиеДанные =  
    Неопределено Тогда
```

```
        Сообщить("Выберете практическое задание.");
```

```
        Возврат;
```

```
    КонецЕсли;
```

```
    ПрактическоеЗадание =
```

```
    ПолучитьЗадачуПрактическоеЗадание(Элементы.ПрактическиеЗадания.Теку  
    щиеДанные.ПрактическоеЗадание);
```

```
    Если ПрактическоеЗадание = Неопределено Тогда
```

ОткрытьФорму("Задача.ПрактическоеЗадание.Форма.ФормаЗадачи",  
Новый Структура("Занятие, ПрактическоеЗадание", Объект.Ссылка,  
Элементы.ПрактическиеЗадания.ТекущиеДанные.ПрактическоеЗадание));

Иначе

ОткрытьФорму("Задача.ПрактическоеЗадание.Форма.ФормаЗадачи",  
Новый Структура("Ключ", ПрактическоеЗадание));

КонецЕсли;

КонецПроцедуры

&НаСервере

Функция ПолучитьЗадачуПрактическоеЗадание(ПрактическоеЗадание)

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| ТаблицаПрактическоеЗадание.Ссылка КАК Ссылка

|ИЗ

| Задача.ПрактическоеЗадание КАК

ТаблицаПрактическоеЗадание

|ГДЕ

| ТаблицаПрактическоеЗадание.Кандидат = &Кандидат

| И

ТаблицаПрактическоеЗадание.ПрактическоеЗадание =

&ПрактическоеЗадание";

Запрос.УстановитьПараметр("Кандидат",

ПараметрыСеанса.ТекущийПользователь);

Запрос.УстановитьПараметр("ПрактическоеЗадание",  
ПрактическоеЗадание);

ВыборкаДетальныеЗаписи = Запрос.Выполнить().Выбрать();

Если ВыборкаДетальныеЗаписи.Следующий() Тогда  
    Возврат ВыборкаДетальныеЗаписи.Ссылка;  
КонецЕсли;

Возврат Неопределено;

КонецФункции

#КонецОбласти

## Приложение Д

### Реализация модуля формы задачи «Тестовое задание»

#Область СобытияФормы

&НаСервере

Процедура ПриСозданииНаСервере(Отказ, СтандартнаяОбработка)

Если Объект.Выполнена Тогда

    ЭтаФорма.Доступность = Ложь;

    Элементы.ФормаПроверитьТест.Доступность = Ложь;

КонецЕсли;

Если НЕ ЗначениеЗаполнено(Объект.Кандидат) Тогда

    Объект.Кандидат =

    ПараметрыСеанса.ТекущийПользователь;

КонецЕсли;

Если Параметры.Свойство("ТестовоеЗадание") Тогда

    Объект.Наименование =

    Параметры.ТестовоеЗадание.Наименование;

    Объект.ТестовоеЗадание = Параметры.ТестовоеЗадание;

КонецЕсли;

Если НЕ ЗначениеЗаполнено(Объект.Наставник)

    И Параметры.Свойство("Занятие") Тогда

        Попытка

            Объект.Наставник =

            РегистрыСведений.ДоступКЗанятиям.Получить(Новый Структура("Занятие,  
Кандидат", Параметры.Занятие, Объект.Кандидат)).Наставник;

Исключение

Отказ = Истина;

Сообщить("Не удалось заполнить параметры формы.

Обратитесь к администратору.");

КонецПопытки;

КонецЕсли;

Если Объект.Ссылка.Пустая() Тогда

ТестовоеЗаданиеПриИзмененииНаСервере();

КонецЕсли;

КонецПроцедуры

&НаКлиенте

Процедура ТестовоеЗаданиеПриИзменении(Элемент)

ТестовоеЗаданиеПриИзмененииНаСервере()

КонецПроцедуры

&НаСервере

Процедура ТестовоеЗаданиеПриИзмененииНаСервере()

Объект.Вопросы.Очистить();

Для каждого СтрокаВопрос Из Объект.ТестовоеЗадание.Вопросы

Цикл

НоваяСтрока = Объект.Вопросы.Добавить();

НоваяСтрока.Вопрос = СтрокаВопрос.Вопрос;

КонецЦикла;

КонецПроцедуры

#КонецОбласти

#Область КомандыФормы

&НаКлиенте

Процедура ПроверитьТест(Команда)

    ПроверитьТестНаСервере();

КонецПроцедуры

&НаСервере

Процедура ПроверитьТестНаСервере()

    Для каждого СтрокаВопрос Из Объект.Вопросы Цикл

        Если СтрокаВопрос.Ответ = СтрокаВопрос.Вопрос.Ответ

Тогда

        СтрокаВопрос.ВерныйОтвет = Истина;

        КонецЕсли;

    КонецЦикла;

Объект.Выполнена = Истина;

Попытка

    Записать();

    ЭтаФорма.Доступность = Ложь;

    Элементы.ФормаПроверитьТест.Доступность = Ложь;

Исключение

    Отказ = Истина;

    Сообщить("Не удалось записать объект. Обратитесь к администратору.");

КонецПопытки;

КонецПроцедуры

#КонецОбласти