

## Содержание

Введение.....	3
1 Разработка в среде Rational Rose.....	4
1.1 Требования заказчика.....	4
1.2 Варианты использования и действующие лица.....	4
1.3 Взаимодействие объектов.....	7
1.4 Диаграмма последовательности.....	7
1.5 Диаграмма кооперативная.....	8
1.6 Классы и пакеты.....	9
1.7 Представление компонентов.....	10
1.8 Представление размещения.....	11
2 Создание БД в системе 1С: Предприятие.....	14
2.1 Описание 1С: Предприятие.....	14
2.2 Создание информационной базы.....	14
2.3 Создание объекта Справочник.....	18
2.3.1 Создание справочника Соискатели.....	19
Заключение.....	21
Список литературы.....	22

## **Введение**

Развитие различных сфер человеческой деятельности на современном этапе невозможно без широкого применения вычислительной техники и создания информационных систем различного направления. Обработка информации в подобных системах стала самостоятельным научно – техническим направлением.

После этапа построения информационной модели начинается проектирование системы. На этом этапе производится выбор технологических решений, на основе которых будет построена информационная система.

Информация в современном мире превратилась в один из наиболее важных ресурсов, а информационные системы (ИС) стали необходимым инструментом практически во всех сферах деятельности.

В реальных условиях проектирование - это поиск способа, который удовлетворяет требованиям функциональности системы средствами имеющихся технологий с учетом заданных ограничений.

Разнообразие задач, решаемых с помощью ИС, привело к появлению множества разнотипных систем, отличающихся принципами построения и заложенными в них правилами обработки информации.

В данном курсовом проекте будет создана «АИС Трудоустройство» при помощи программных продуктов Rational Rose и 1С: Предприятие.

## **1 Разработка в среде Rational Rose**

### **1.1 Требования заказчика**

АИС Трудоустройство. Разработать информационную систему для администрирования резюме.

Rational Rose — мощный инструмент анализа и проектирования объектно-ориентированных программных систем. Он позволяет моделировать системы до написания кода, так что вы можете с самого начала быть уверены в адекватности их архитектуры. С помощью готовой модели недостатки проекта легко обнаружить на стадии, когда их исправление не требует еще значительных затрат.

Среда Rational Rose позволяет проектировать варианты использования и их диаграммы для визуализации функциональных возможностей системы. Диаграммы Взаимодействия показывают, как объекты работают совместно, обеспечивая требуемые функциональные возможности. Для отображения объектов системы и их отношений используются диаграммы Классов. Диаграммы Компонентов иллюстрируют, как классы соотносятся с готовыми физическими компонентами системы. Наконец диаграммы Размещения применяют для визуализации проекта распределенных систем.

Модель Rose — это картина системы. Она содержит все диаграммы UML, действующих лиц, варианты использования, объекты, классы, компоненты и узлы системы. Она детально описывает, что система содержит и как функционирует, поэтому разработчики могут использовать ее в качестве эскиза или чертежа создаваемой системы.

### **1.2 Варианты использования и действующие лица**

Представление Вариантов Использования, как правило, не зависит от

реализации модели. Варианты использования и действующие лица описывают сферу применения проекта (project scope), но не вникают в такие детали его реализации, как, например, используемый язык программирования.

Одним из основных преимуществ применения диаграммы Вариантов Использования является то, что она предоставляет важную информацию. Взглянув на варианты использования, ваши клиенты поймут, какие функциональные возможности будут заложены в систему. Рассматривая действующих лиц, они выяснят, кто конкретно будет с ней взаимодействовать. Изучая все множество вариантов использования и действующих лиц, они определяют сферу применения системы, что она должна будет делать. Это поможет им узнать также, что она не будет делать, и внести коррективы. Например, взглянув на диаграмму, пользователь может сказать: "Все это прекрасно, но я хочу иметь еще возможность получать отчет о десяти последних транзакциях для моего счета".

Конкретная цель диаграмм Вариантов Использования — документирование вариантов использования (все входящее в сферу применения системы), действующих лиц (все вне этой сферы) и связей между ними. Разрабатывая диаграммы Вариантов Использования для системы трудоустройства представлена на рис.1.

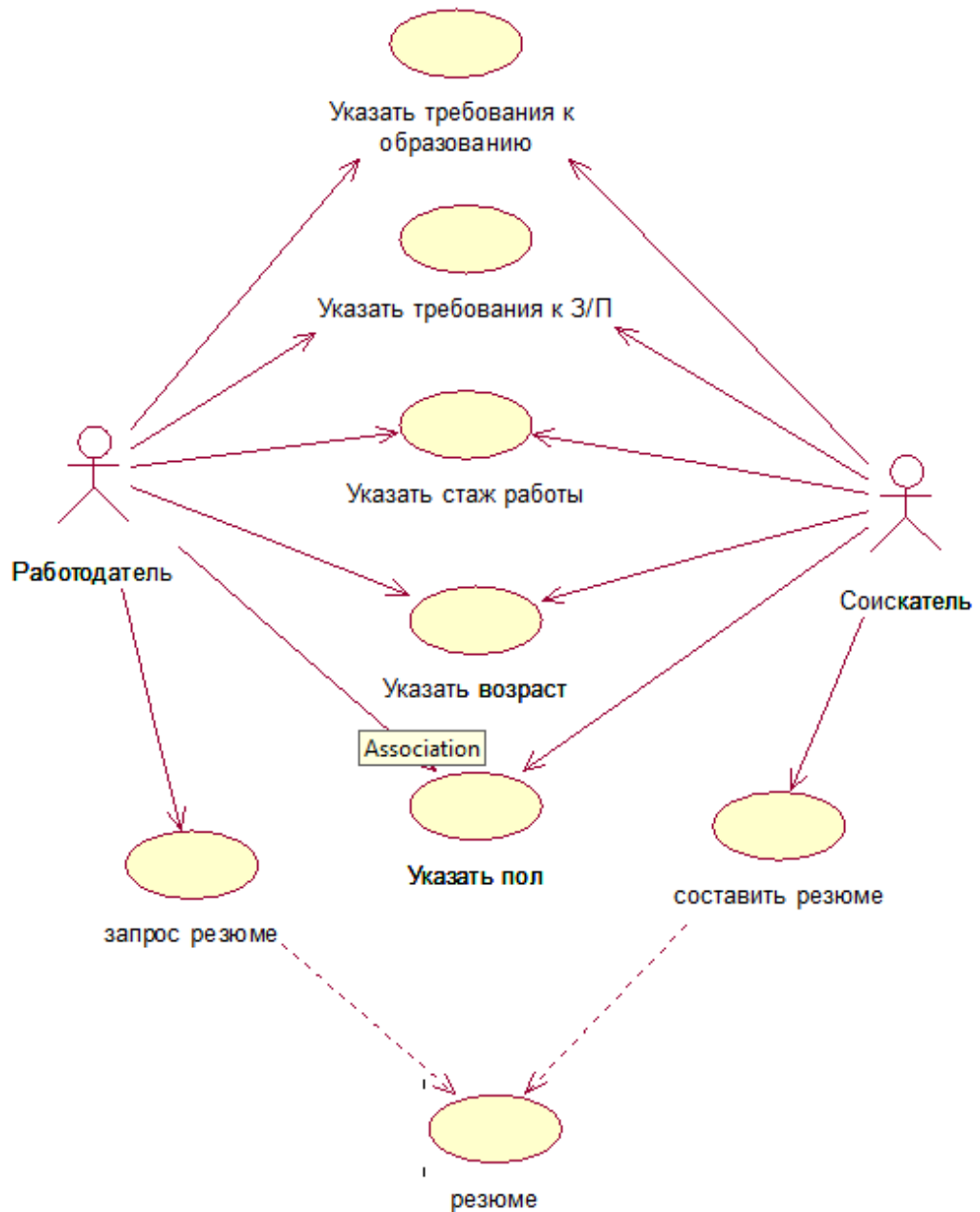


Рис. 1 - Диаграмма вариантов использования

Диаграмма вариантов использования показывает, основные функции системы (купить книгу, взять книгу, регистрация и т.д.), взаимодействие объектов при выполнении вышеперечисленных функций. Что даёт общее представление о системе.

### 1.3 Взаимодействие объектов

Существуют два типа диаграмм Взаимодействия: диаграммы Последовательности и Кооперативные диаграммы. Оба отображают события, участвующие в процессе обработки информации варианта использования, и сообщения, которыми обмениваются объекты. События на диаграмме Последовательности упорядочены по времени, а Кооперативная диаграмма организована вокруг самих объектов.

С помощью диаграмм Взаимодействия проектировщики и разработчики системы могут определить классы, которые нужно создать, связи между ними, а также операции и ответственности (responsibilities) каждого класса. Диаграммы Взаимодействия — краеугольный камень, на котором возводится оставшаяся часть проекта.

Диаграммы Взаимодействия содержат:

Объекты: Можно использовать имена как объектов, так и классов, или того и другого.

Сообщения: С помощью сообщения один объект или класс запрашивает у другого выполнения какой-то конкретной функции. Например, форма может запросить у объекта Отчет напечатать ее.

### 1.4 Диаграмма последовательности

Диаграммы Последовательности упорядочены по времени. Они полезны для того, кто хочет понять логическую последовательность событий в сценарии. Хотя информация о последовательности входит и в Кооперативные диаграммы, она лучше воспринимается на диаграмме Последовательности.

Составим для разрабатываемой системы диаграмму последовательности.

«Соискатель пишет резюме»

Соискатель, желающий устроиться на работу в определённую организацию составляет резюме. Диаграмма последовательности для этого

случая представлена на рис. 2. Выделяются пять объектов: Соискатель, Резюме, БД, Администратор БД и Организация. Стрелками показаны взаимодействия этих объектов, а именно пересылка сообщениями между этими объектами, необходимых для выполнения поставленной задачи.

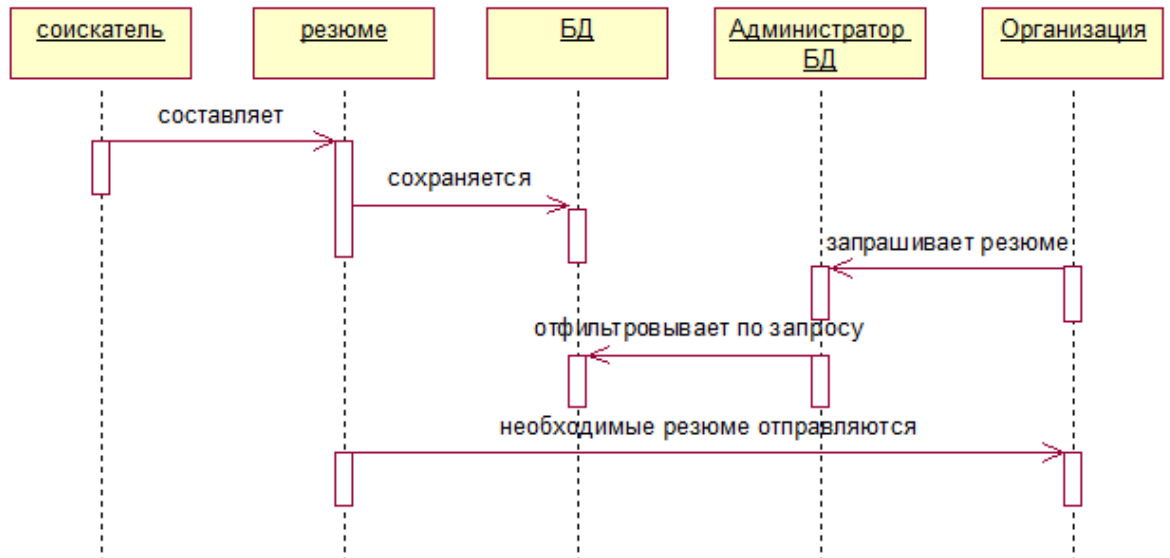


Рис. 2 – Диаграмма последовательности

### 1.5 Диаграмма кооперативная

Кооперативные диаграммы полезны в тех случаях, когда нужно оценить последствия сделанных изменений. Кооперативная диаграмма показывает, какие объекты взаимодействуют друг с другом. При внесении изменений в объект становится понятно, на какие другие объекты это повлияет.

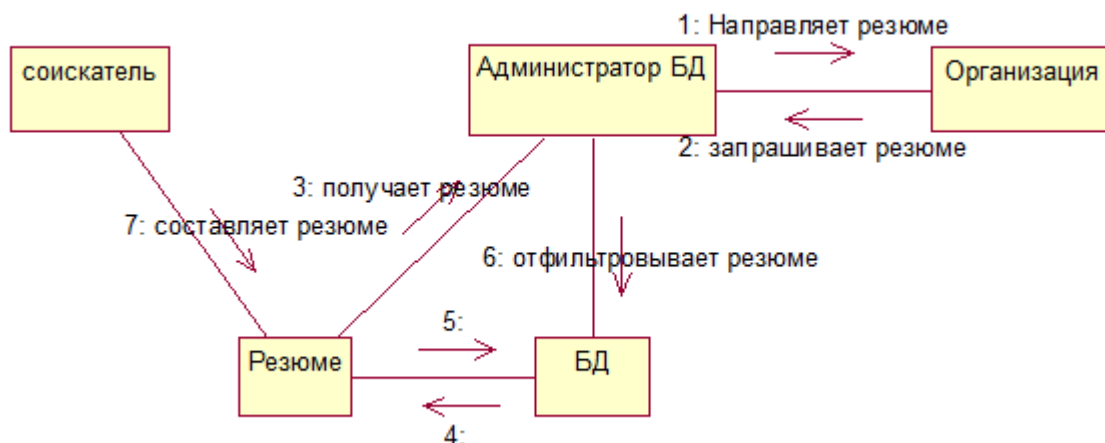


Рис. 3 – Диаграмма сотрудничества

### 1.6 Классы и пакеты

Объектом называют нечто, заключающее (инкапсулирующее) в себе некоторые данные и поведение. Это термин, описывающий реальные, конкретные предметы. Данные объекта называются атрибутами (attributes). Поведение объекта представляется его операциями (operations). В среде Rose объекты помещают на диаграммы Взаимодействия. Когда действующее лицо (представляющее собой стереотип класса) или какой-то другой класс переносится на диаграмму Взаимодействия, автоматически создается экземпляр объекта этого класса. Удаление объекта с диаграммы Rose не приводит к удалению класса из модели.

Класс — это некая сущность, представляющая собой как бы схему объекта. Иными словами, класс определяет данные и поведение, которыми должен обладать объект.

На языке UML такие элементы, как действующие лица, варианты использования, классы и компоненты, можно сгруппировать в пакеты (packages). В частности, в представлении Вариантов Использования можно сгруппировать в пакеты варианты использования и действующих лиц

В разрабатываемой системе, не было особого смысла создавать пакеты, т.к. объём информации достаточно маленький. Основные классы



представлены на рис.4. Построив диаграммы Взаимодействия объектов, были выявлены основные операции классов и составлены необходимые атрибуты, и соответствующие атрибутам типы данных.

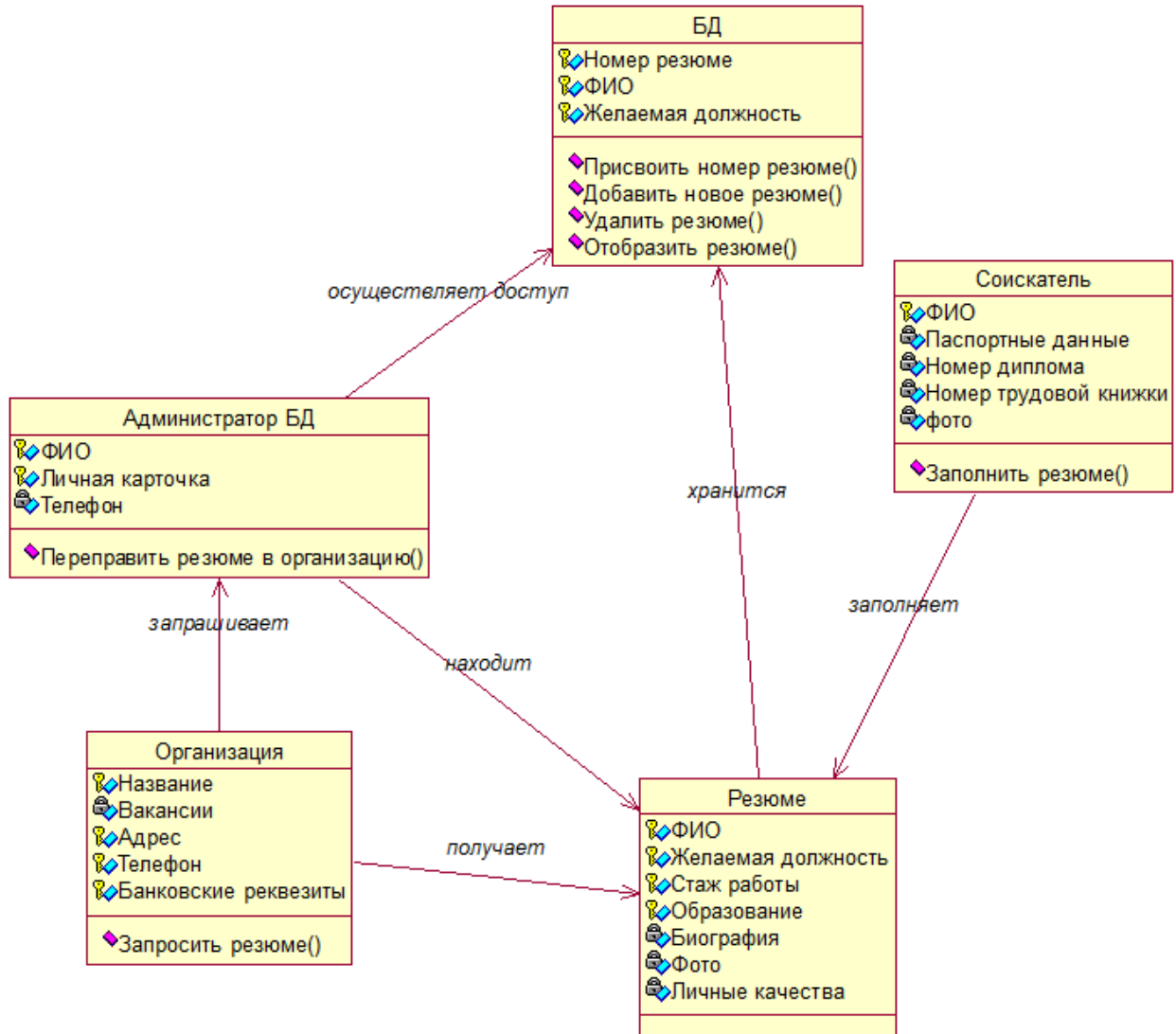


Рис. 4 – Диаграмма классов

### 1.7 Представление компонентов

Компонентом (component) называется физический модуль кода. Компонентами бывают как библиотеки исходного кода, так и исполняемые файлы. Например, если вы работаете на языке C++, то файлы .CPP и .H будут отдельными компонентами. Получающийся при компиляции исполняемый .EXE файл также является компонентом системы.

Перед началом генерации кода необходимо соотнести каждый из файлов с соответствующими компонентами. На языке C++ каждый класс соотносится с двумя компонентами, один из которых соответствует .CPP файлу этого класса, а другой — .H файлу.

На рис.5 представлен общий вид диаграммы компонентов, состоящей из основного исполняющего файла (main.exe) взаимодействующего с драйверами базы данных, дополнительными библиотеками и компонентами самой базы данных.

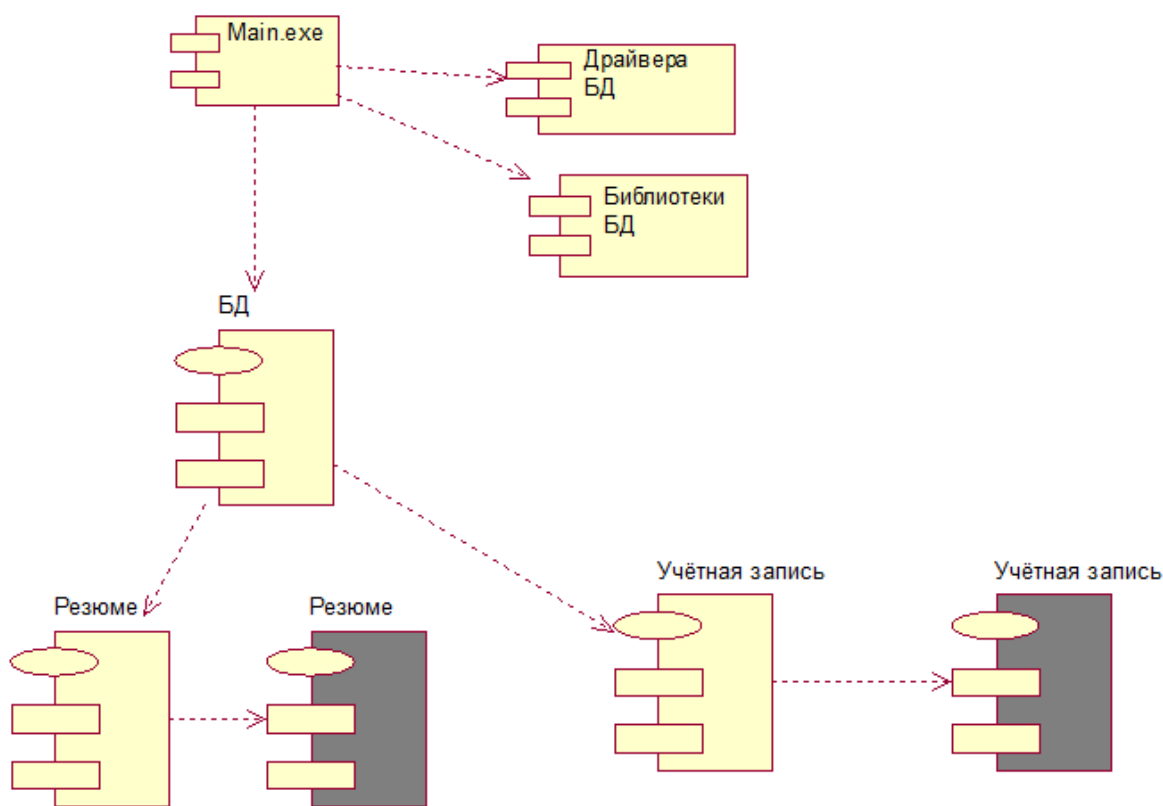


Рис.5 – Диаграмма компонентов

### 1.8 Представление размещения

Представление Размещения (Deployment: view). Оно отражает физическое распределение готового приложения, включая размещение и топологию сети, а также локализацию в ней компонентов системы.

Рассматриваются и такие проблемы, как определение требуемой полосы пропускания сети, предполагаемого количества параллельно работающих пользователей, действий при неполадках на сервере и т.д.

Представление Размещения содержит процессоры, устройства, процессы и связи между процессорами и устройствами. Все они наносятся на диаграмму Размещения (Deployment diagram). Для системы и, следовательно, для модели Rose может быть создана только одна диаграмма Размещения.

Диаграмма размещения для разрабатываемой системы представлена на рис.6. Сервер БД является отдельной машиной на которой стоит База данных. По средствам сети сервер соединяется с рабочей станцией, на которой установлена программа main.exe, которая производит все необходимые операции, обращаясь лишь к серверу за информацией из БД. Далее информация передаётся на вторую рабочую станцию, где резюме открывается при помощи обычного текстового редактора. При необходимости с этой рабочей станции происходит печать при помощи подключенного к ней лазерного принтера.

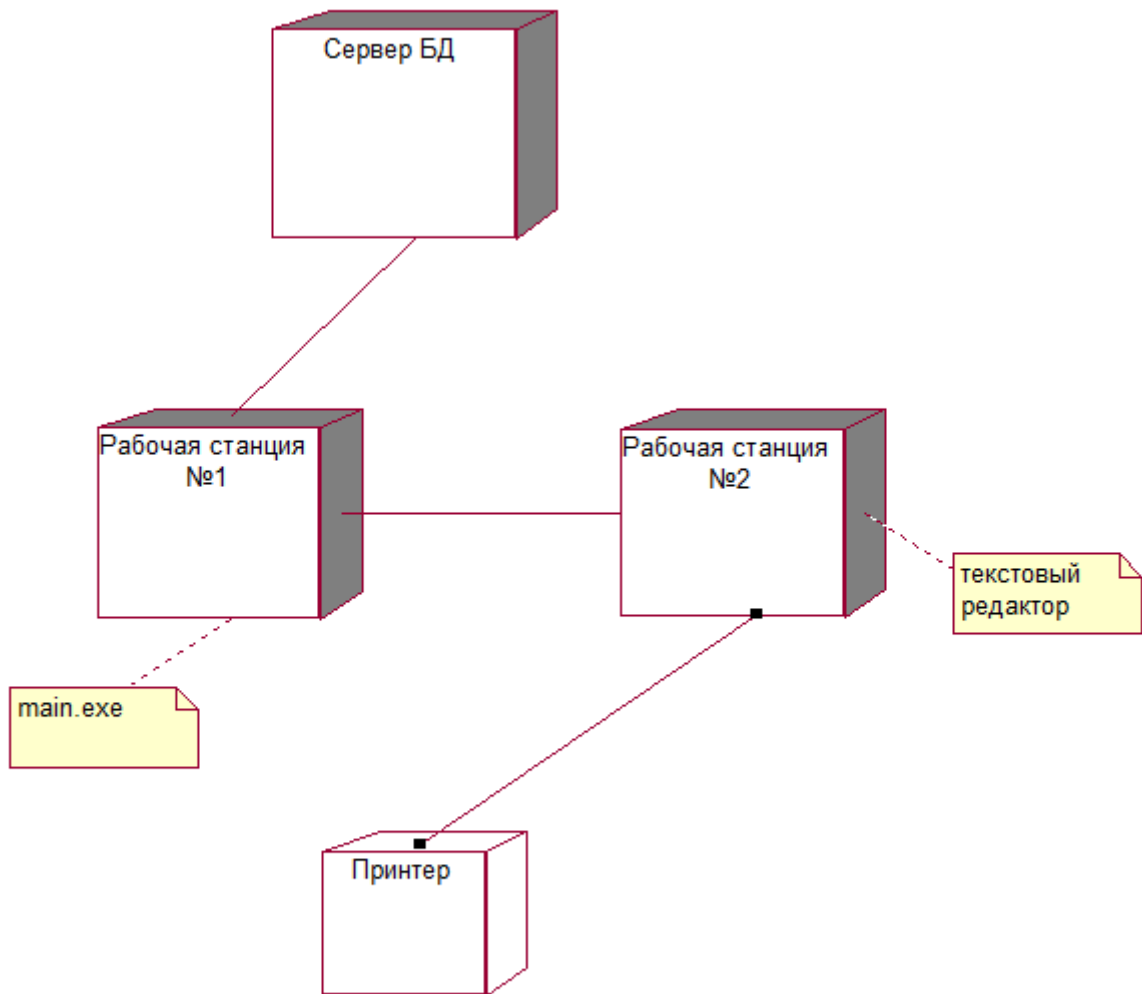


Рис.6 – Диаграмма размещения

## 2 Создание БД в системе 1С: Предприятие

### 2.1 Описание 1С: Предприятие

Конфигуратор системы 1С:Предприятие является специальным режимом запуска 1С и предназначен для разработчиков, программистов. В этом режиме разработчик создает или корректирует структуру базы данных, программные модули, производит административные работы. В режиме Конфигуратора нельзя вводить информацию в базу данных, за исключением загрузки базы данных из архивов. Обычные пользователи работают в другом режиме «Предприятие». В этом режиме возможен ввод данных, заполнение справочников и ввод документов, формирование отчетов и запуск обработок. В режиме Предприятие нельзя редактировать структуру конфигурации.

Система 1С:Предприятие позволяет вести практически неограниченное количество необходимых справочников. Каждый справочник представляет собой список однородных объектов: сотрудников, организаций, товаров и т. д. Каждый такой объект называется *элементом справочника*. С точки зрения пользователя, следует иметь в виду, что в **Конфигураторе** создается не сам справочник, как список значений, а разрабатывается заготовка справочника, его шаблон.

Объект конфигурации Документ является прикладным объектом и предназначен для описания информации о совершенных хозяйственных операциях или о событиях, произошедших в жизни организации вообще. На основе объекта конфигурации Документ платформа создает в базе данных информационную структуру.

Документ обладает способностью проведения. Факт проведения документа означает, что событие, которое он отражает, повлияло на состояние учета. До тех пор, пока документ не проведен, состояние учета неизменно.

### 2.2 Создание информационной базы

Система **1С:Предприятие 8.2** может работать с некоторым количеством учетных баз в режиме конфигуратора и в режиме предприятия (в зависимости от

режима лицензирования). Для начала работы создадим информационную базу для ведения учета. При запуске системы производится выбор информационной базы и режима запуска системы (рис.7).

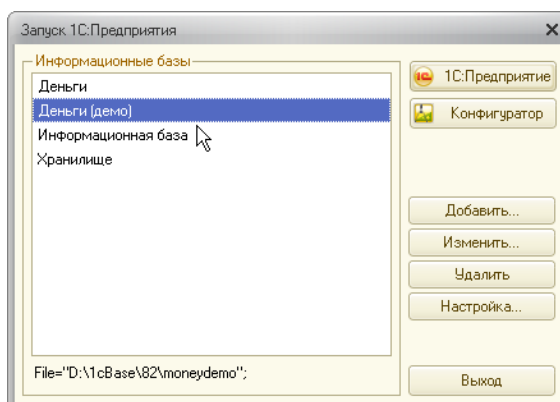


Рис. 7 - Диалоговое окно запуска системы

Для добавления новой базы следует нажать кнопку «Добавить». Затем в окне мастера (рис.8) необходимо выбрать «Создание новой информационной базы».

Установите переключатель в положение «Создание новой информационной базы». И нажать кнопку «Далее».

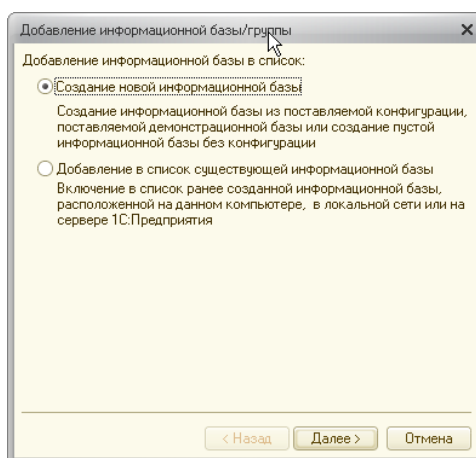


Рис. 8 - Выбор вида информационной базы

На следующем экране мастера установить переключатель в положение «Создание информационной базы без конфигурации для разработки новой конфигурации или загрузки выгруженной ранее конфигурации». И нажать кнопку «Далее».

На следующем экране указать наименование информационной базы и тип расположения (рис.9). И нажать кнопку «Далее».

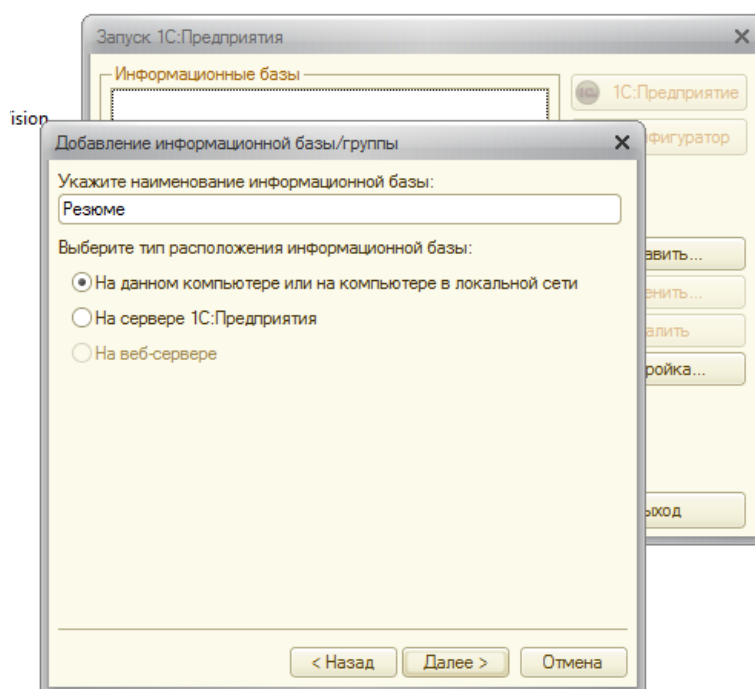


Рис. 9 - Информация о наименовании базы и ее расположении

На последнем экране выбрать место хранения информационной базы и язык интерфейса (рис. 10). И нажать кнопку «Далее».

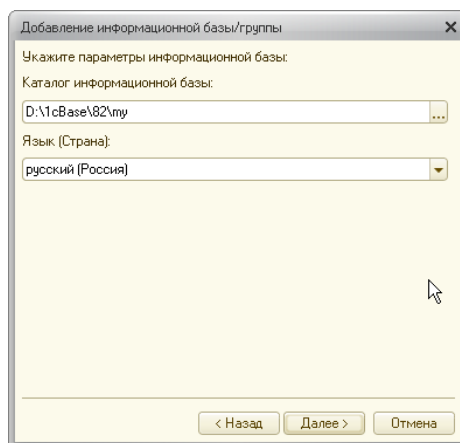


Рис. 10. Выбор места хранения базы и языка интерфейса

После завершения процедуры добавления информационной базы в списке окна запуска появится новая строка с реквизитами новой информационной базы.

Запустите вашу информационную базу в режиме «Конфигуратора». Выберите в меню «**Конфигурация**» пункт «**Открыть конфигурацию**». На экране откроется «**Дерево конфигурации**» (рис. 11).



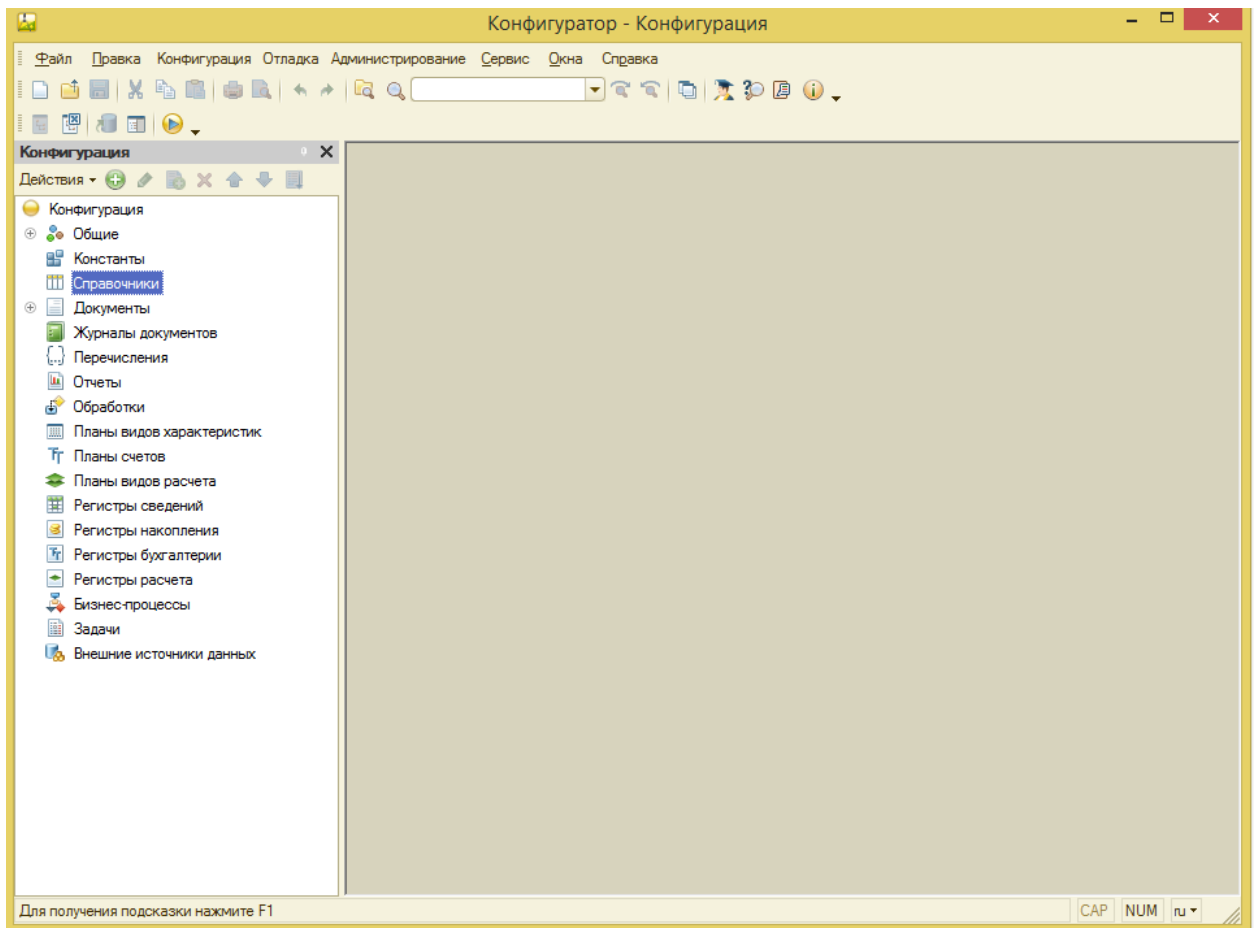



Рис.11 - Дерево конфигурации

После того, как конфигурация открыта, ее состав появляется в окне дерева конфигурации. Это окно вы можете закрыть, как любое другое окно Windows, при этом конфигурация останется открытой (т.е. доступной для редактирования). Чтобы снова отобразить на экране окно дерева конфигурации следует воспользоваться командой меню **Конфигурация - Окно конфигурации**.

### 2.3 Создание объекта Справочник

Создайте новый объект конфигурации. Создать новый объект конфигурации можно несколькими способами:

- установите курсор на тот вид объектов конфигурации, который вас интересует, и в командной панели окна конфигурации нажмите кнопку «Действия» – «Добавить» или воспользуйтесь кнопкой панели инструментов .

- воспользоваться контекстным меню, которое вызывается при нажатии на правую клавишу мыши. Установите курсор на вид объектов конфигурации и нажмите правую клавишу мыши. В появившемся меню выберите пункт «Добавить».

### 2.3.1 Создание справочника Соискатели

Данный справочник будет содержать имена членов семьи и названия сторон, от имени которых производятся доходы и расходы, и кому принадлежат кошельки. После нажатия «Добавить» система откроет окно редактирования объекта конфигурации.

Задайте имя справочника – «Соискатели». На основании имени платформа автоматически создаст синоним – «Соискатели»

На вкладке «Данные» установите значения кода и наименования. Уменьшите длину кода до 2 и увеличьте длину наименования до 50 (рис 12).

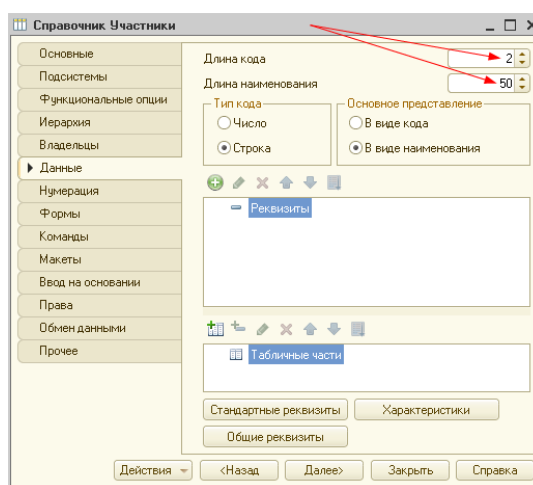


Рис. 12 - Справочник Участники. Данные.

Все остальные свойства объекта конфигурации «Справочник» оставьте по умолчанию, и нажмите «Закреть».

Готовая база представлена на рисунке 13.

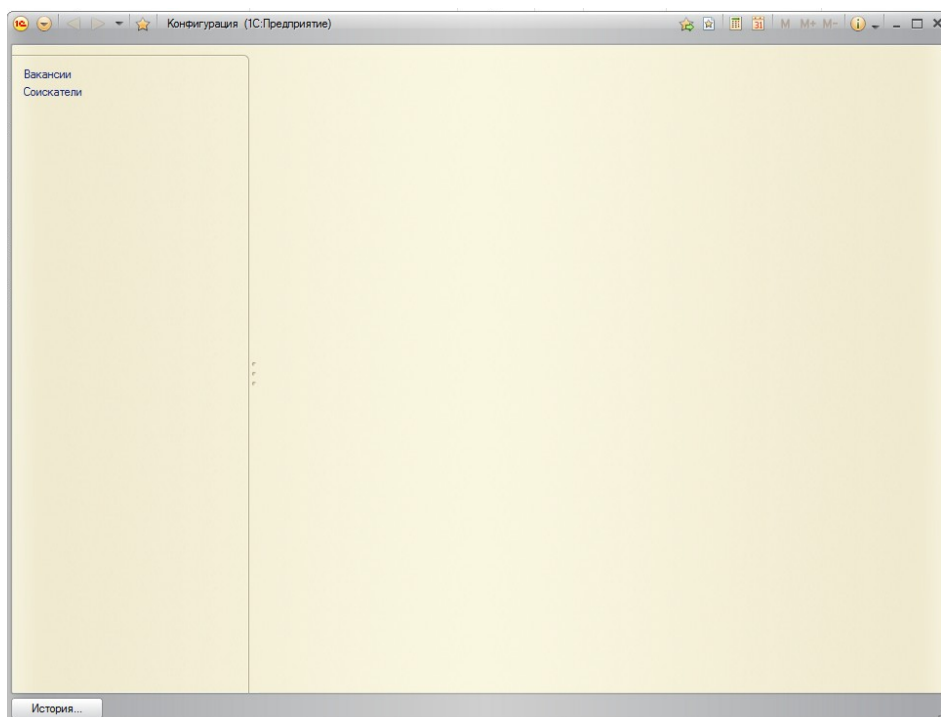


Рис. 13 – Готовая база

## Заключение

Проектирование информационной системы - это один из важнейших этапов ее существования то, с чего, собственно, должна начинаться её жизнь. Таким образом, прежде чем проектировать сеть, нужно понять, какие задачи будет решать сеть, какими будут основные потоки трафика, как физически будут расположены пользователи и ресурсы, нужно ли задание приоритетов видов трафика, как будут решаться вопросы защиты информации внутри сети, как сеть будет подключена к Интернет, как решить задачи управления правами доступа пользователей. Кроме того, в задачу предпроектного исследования входит изучение состояния зданий и сооружений в месте развертывания сети, анализ существующей инфраструктуры. Эта информация жизненно необходима как для постановки задачи проектирования, так и для самого проектирования.

В данной курсовой работе была разработана информационная система при помощи диаграмм UML и создана БД в среде 1С: Предприятие 8.2.

## Список литературы

1. А.Н. Набатов, Методические указания к выполнению лабораторных работ, Уфа.
2. Грекулов В.И., Денищенко Г.Н., Коровкина Интернет – университет информационных технологий – ИНТУИТ. ру, 2008г, 304 стр
3. Intuit.ru