

1.1.1 Модель движения потоков данных в стандарте IDEF0 с дополнительными уровнями декомпозиции

IDEFO – методология функционального моделирования [8]. С помощью простого и гармоничного графического языка IDEF0, моделирования система предстает перед разработчиками и аналитиками в виде набора взаимосвязанных функциональных блоков.

Процесс моделирования деятельности бухгалтера в методологии IDEF0 представлена на рисунке 1.

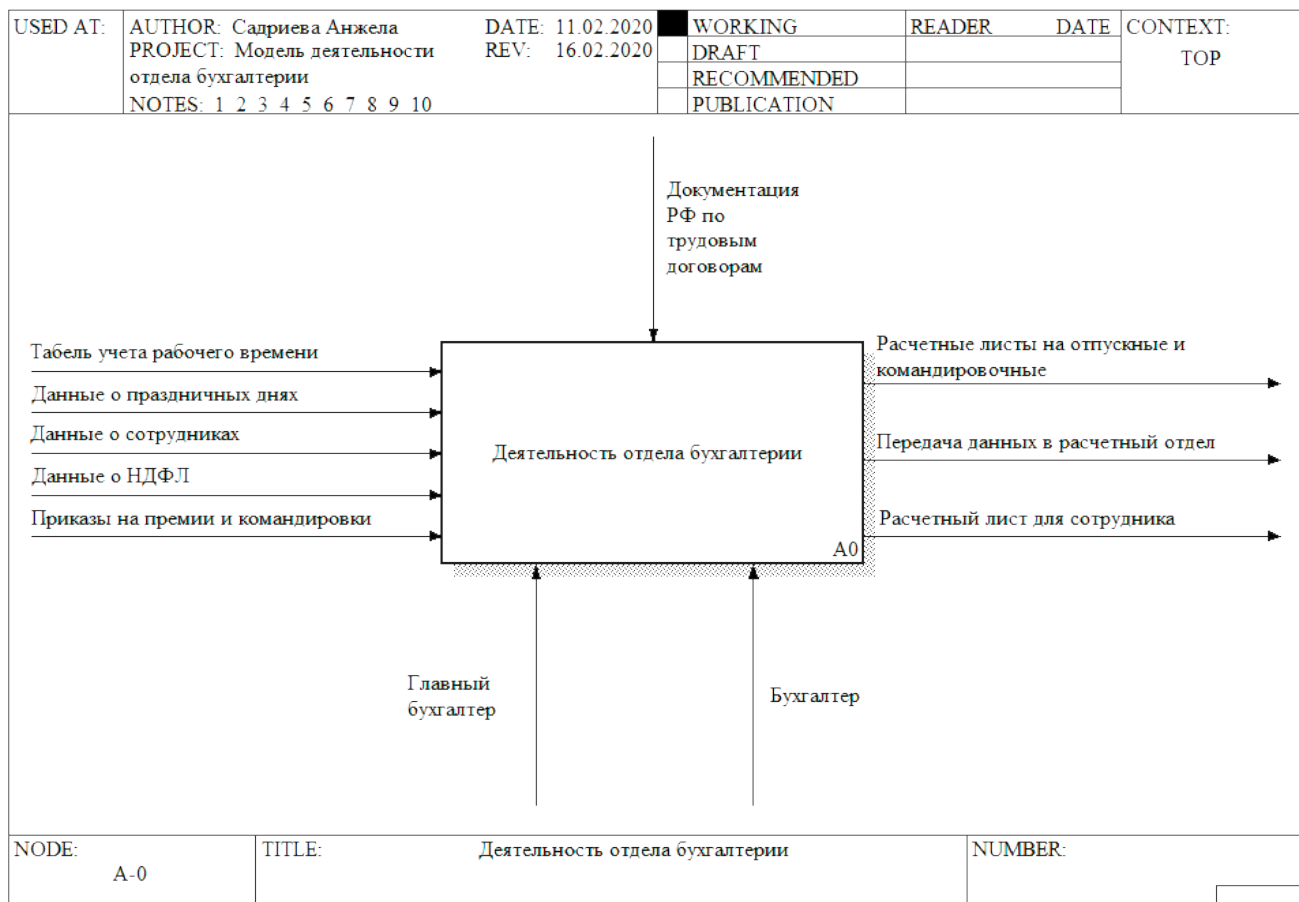


Рисунок 1 Контекстная диаграмма деятельности бухгалтера в методологии IDEF0

На диаграмме действующими лицами (актерами) являются бухгалтер и главный бухгалтер, главная задача которых – расчет заработных плат работников в ПАО «Таттелеком бухгалтером и проверка данных расчетов, и необходимых документаций главным бухгалтером. Входными данными являются: табель учета рабочего времени, данные о праздничных днях, о сотрудниках и о НДФЛ (13%), документация РФ по трудовым договорам, а также приказы на премии и командировки.

					ЛНТ 0. 09.02.04 17 02. ПЗ		
Изм.	Лист	№ докум.	Подп.	Дата			
Разраб.		Садриева А.Ф.			Проектирование АИС расчета заработной платы работников в ПАО «Таттелеком»		
Пров.							
Н. контр.							
Утв.							
					Лит.	Лист	Листов
						7	43
					ИС-16		

Выходные данные: расчетные листы на отпускные и командировочные, передача данных в расчетный отдел, расчетный лист для сотрудника.

Декомпозиция (Decomposition) является основным понятием стандарта IDEF0 [9]. Принцип декомпозиции применяется при разбиении сложного процесса на составляющие его функции. При этом уровень детализации процесса определяется непосредственно разработчиком модели.

Декомпозиция позволяет постепенно и структурировано представлять модель системы в виде иерархической структуры отдельных диаграмм, что делает ее менее перегруженной и легко усваиваемой.

На рисунке 2 представлена декомпозиция деятельности бухгалтерии первого уровня в стандарте IDEF0, в которой рассматриваются 3 бизнес-процесса: изучение данных о сотрудниках, подсчет отработанного времени и расчет заработной платы работников и иных выплат, с учетом НДФЛ (13%).

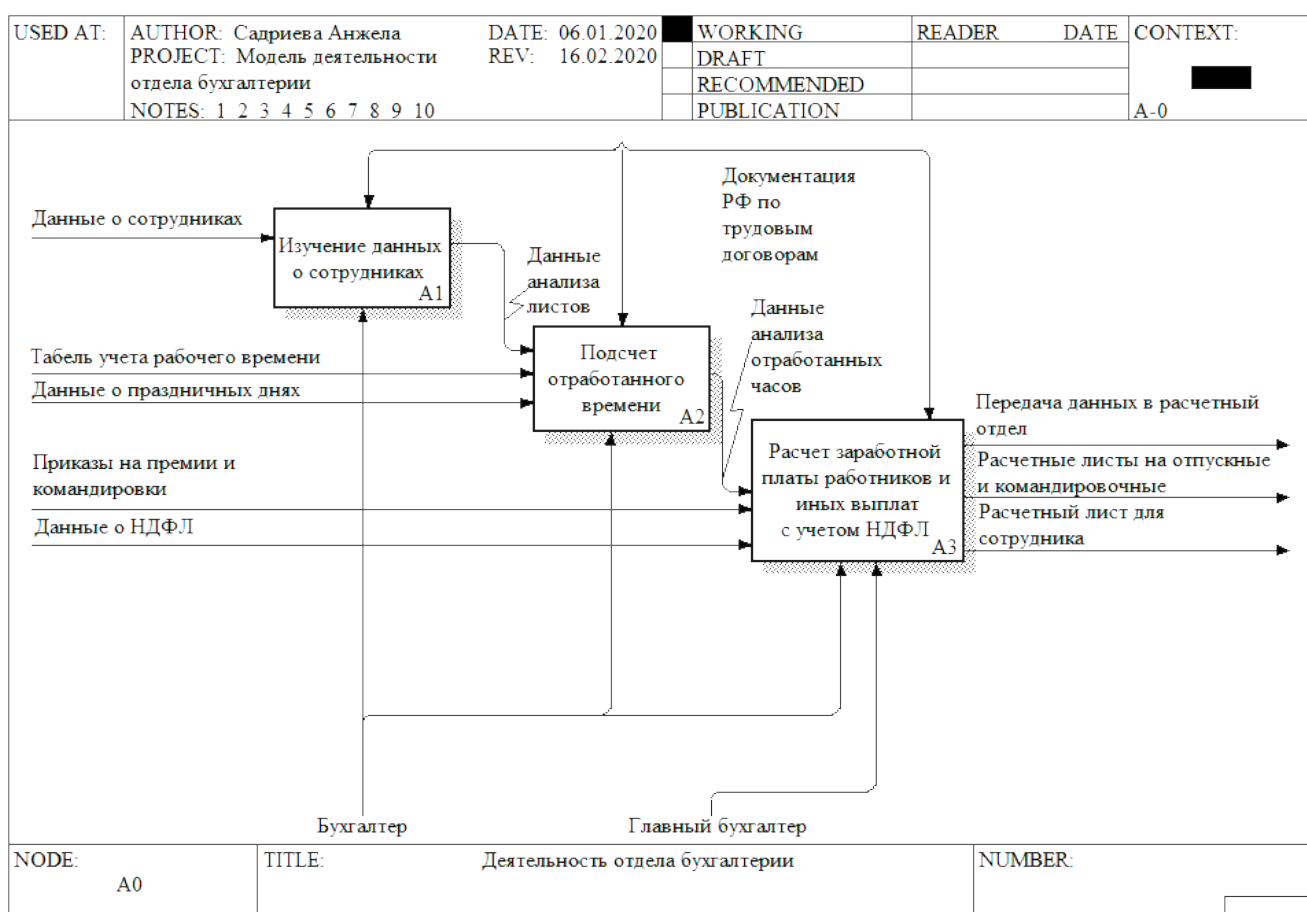


Рисунок 2 Декомпозиция деятельности бухгалтерии первого уровня в стандарте IDEF0

На диаграмме рассматривается процесс при анализе данных о сотрудниках и данных табельных листов. А также показаны действия бухгалтера и главного бухгалтера.

На главного бухгалтера в основном возлагается рассмотрение анализа отработанных часов и информации о сотрудниках, проверка рассчитанных данных заработной платы работников в ПАО «Таттелеком».

Бухгалтер же занимается процессом как изучения данных о сотрудниках, так и подсчета последующего расчета заработной платы работников.

На рисунке 3 представлена декомпозиция деятельности бухгалтерии второго уровня в стандарте IDEF0 блока А3 «Расчет заработной платы работников и иных выплат, с учетом НДФЛ».

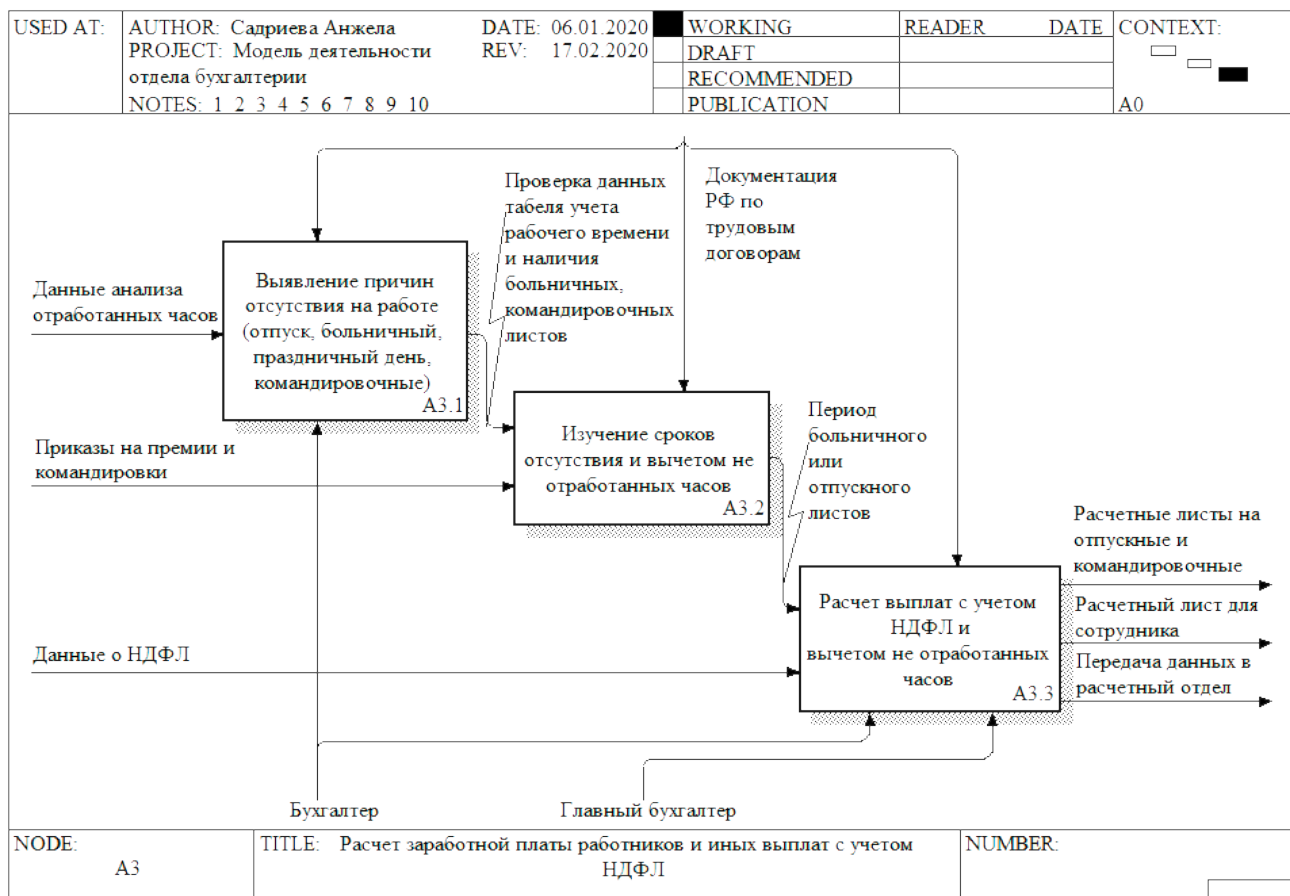


Рисунок 3 Декомпозиция деятельности бухгалтерии второго уровня в стандарте IDEF0

При анализе декомпозиции второго уровня в стандарте IDEF0 можно заметить 3 основных бизнес-процесса, таких как: выявление причин отсутствия на работе (отпуск, больничный, командировочные, праздничный день), изучение сроков отсутствия и расчет выплат с учетом НДФЛ и вычетом не отработанных часов. Бухгалтер выполняет основные функции расчетов с учетом данных отработанного времени и вычета НДФЛ, главный бухгалтер проверяет в основном достоверность больничных, отпускных и командировочных листов, а также просмотра данных табелей учета рабочего времени. Входными данными являются: данные анализа отработанных часов, данные о НДФЛ, документация РФ по трудовым договорам, а также приказы на премии и командировки. Выходные данные: расчетные листы на отпускные и командировочные, передача данных в расчетный отдел, расчетный лист для сотрудника.

1.1.2 Моделирование потоков данных (процессов) в стандарте DFD

Диаграммы потоков данных (Data Flow Diagrams – DFD) представляют собой иерархию функциональных процессов, связанных потоками данных [10].

Каждый процесс должен иметь хотя бы один вход и один выход. Смысл процессов здесь заключается в обработке именно данных, а потому процесс должен получить данные (входящая стрелка) и отдать куда-то после обработки (исходящая стрелка).

Процесс обработки данных должен иметь внешнюю входящую стрелку (данные от внешней сущности) [11]. Для того, чтобы любой подобный процесс начал работать, мало использовать данные из хранилища, должна поступить новая информация для последующей обработки.

На рисунке 4 представлена модель потоков данных в стандарте DFD, в которой представлена деятельность бухгалтерии.

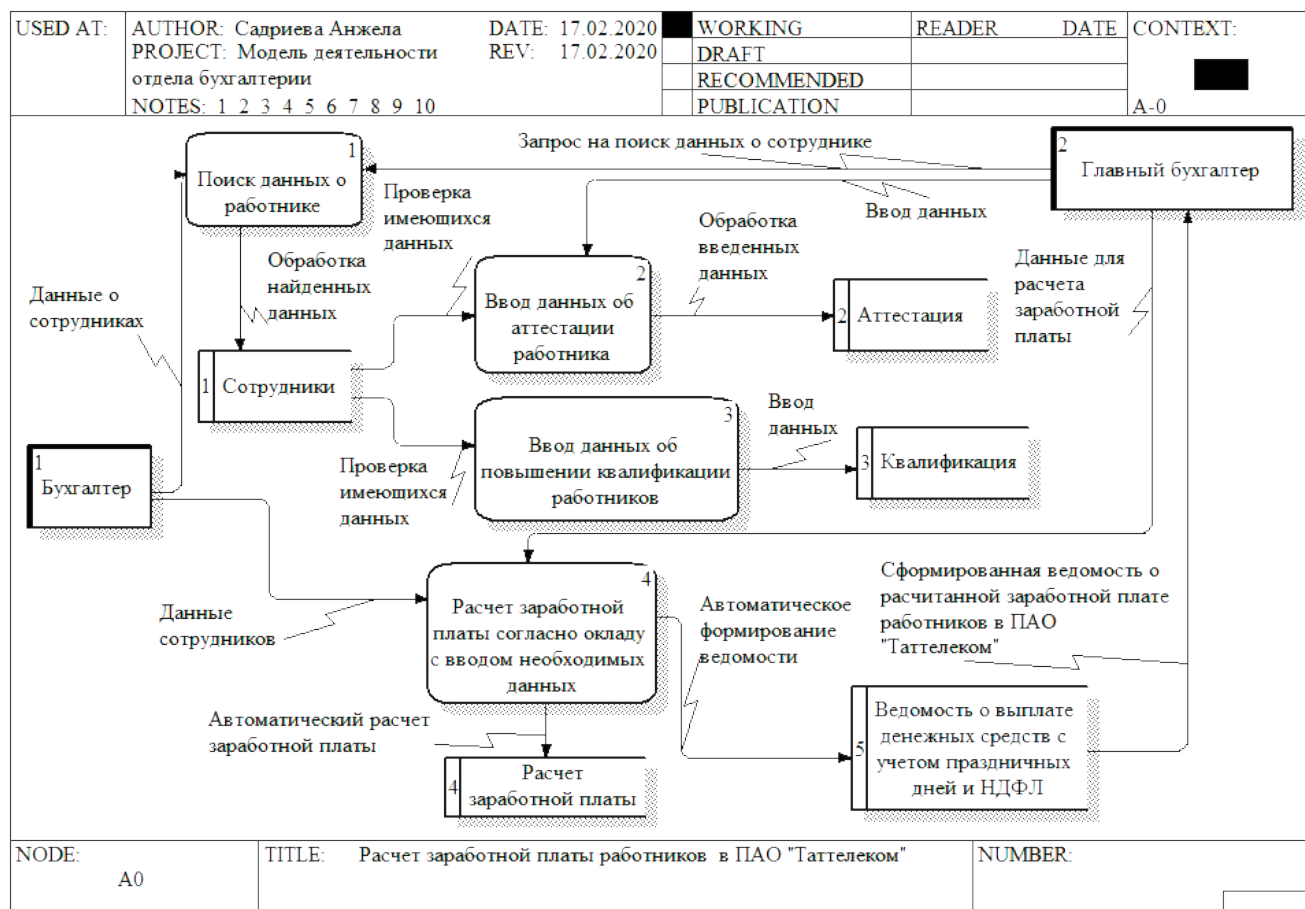


Рисунок 4 Модель потоков данных деятельности бухгалтерии в стандарте DFD

Рассматривая диаграмму DFD можно сказать, что внешними сущностями являются: бухгалтер и главный бухгалтер, расписаны основные процессы деятельности, то есть: изучение данных о работниках и их табелей учета рабочего времени, а также проверка получения приказов

о назначении премии или командировок, составление документации (ведомости, расчетные листы, расчет итоговой заработной платы работников в ПАО «Таттелеком».

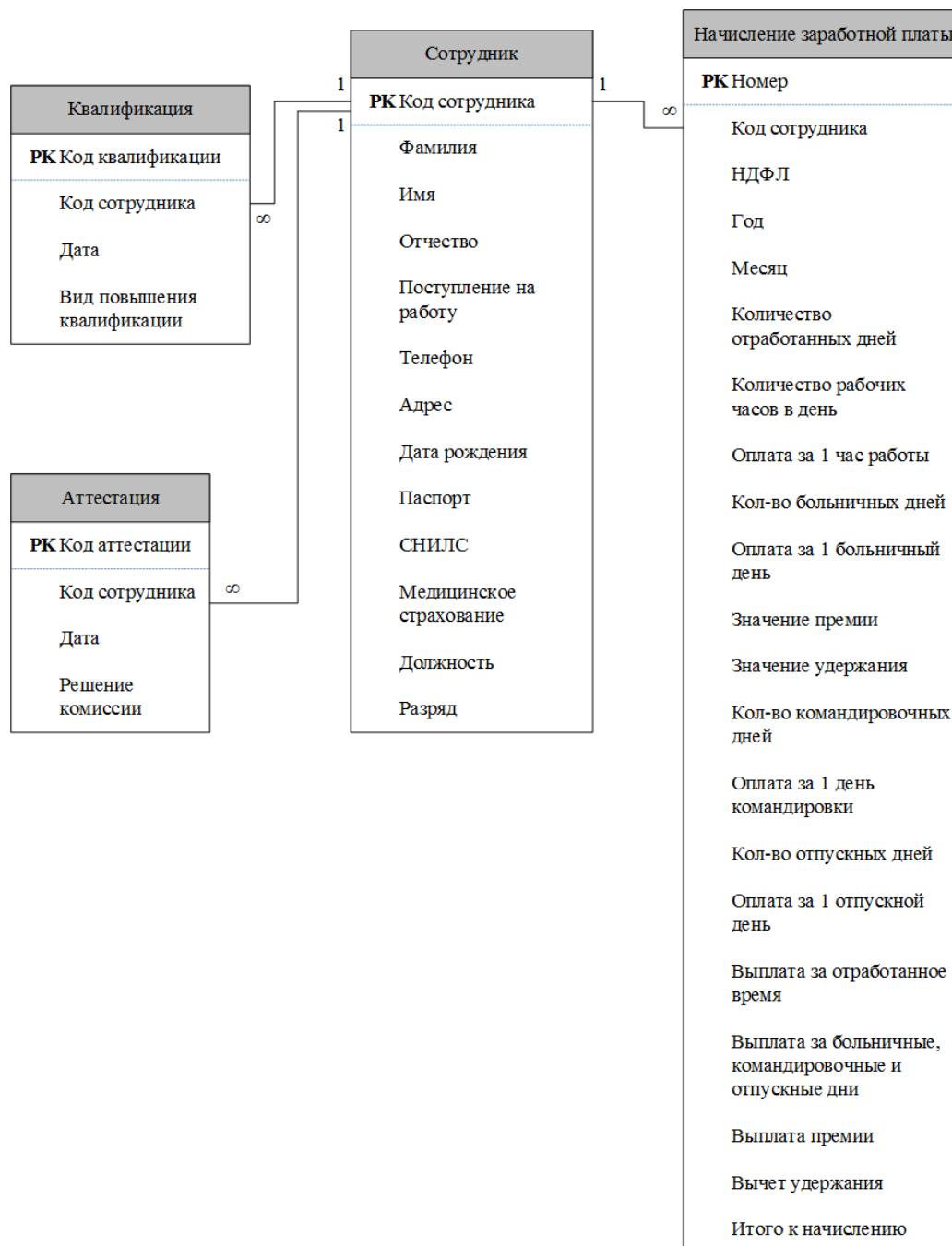
1.1.3 Диаграмма «Сущность-связь» (ERD)

Диаграммы сущность – связь (ERD) предназначены для разработки моделей данных и обеспечивают стандартный способ определения данных и отношений между ними [11].

Сущность представляет собой множество экземпляров реальных или абстрактных объектов, обладающих общими атрибутами или характеристиками [12]. Любой объект системы может быть представлен только одной сущностью, которая должна быть уникально идентифицирована. Имя сущности должно быть существительным, отражающим название типа или класса, а не конкретного экземпляра.

ERD – диаграмма деятельности бухгалтерии представлена на рисунке 5.

					<i>ЛНТ 0. 09.02.04 17 02. ПЗ</i>	<i>Лист</i>
						<i>11</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>		



На ERD диаграмме представлено 4 сущности: Сотрудник, Расчет заработной платы, Аттестация, Квалификация. В них содержится вся необходимая информация для работы с АИС.

1.2 Объектно-ориентированное проектирование

Объектно-ориентированное проектирование представляет собой стратегию, в рамках которой программная система состоит из взаимодействующих объектов, имеющих собственное локальное состояние и способных выполнять определенный набор операций, определяемый состоянием объекта. Объекты скрывают информацию о представлении состояний и, следовательно, ограничивают к ним доступ [13]. Под процессом ООП подразумевается

проектирование классов объектов и взаимоотношений между этими классами [14].

Объектно-ориентированные системы можно рассматривать как совокупность автономных и в определенной степени независимых объектов [15]. Изменение реализации какого-нибудь объекта или добавление новых функций не влияет на другие объекты системы.

Основными задачами построения различного рода диаграмм являются:

- определение вариантов использования и наличия набора функций АИС;
- проектирование и идентификация основных объектов системы;
- разработка модели архитектуры АИС расчета заработной платы работников в ПАО «Таттелеком»;
- определение и документирование интерфейсов объектов.

На основе анализа предметной области при применении объектно-ориентированного подхода будут построены:

- диаграмма вариантов использования;
- диаграмма последовательности;
- логическая модель информационной системы – диаграмма классов;
- модель на физическом уровне – диаграмма компонентов.

1.2.1 Диаграмма вариантов использования

Диаграмма вариантов использования (прецедентов) является исходным концептуальным представлением системы в процессе ее проектирования и разработки.

Данная диаграмма состоит из актеров, вариантов использования и отношений между ними [16]. При построении диаграммы могут использоваться также общие элементы нотации: примечания и механизмы расширения.

Действующее лицо – это роль, которую пользователь играет по отношению к системе.

Действующие лица представляют собой роли, а не конкретных людей или наименования работ.

Действующие лица делятся на три основных типа:

- пользователи;
- системы;
- другие системы, взаимодействующие с данной;
- время.

Время становится действующим лицом, если от него зависит запуск каких-либо событий в системе.

					ЛНТ О. 09.02.04 17 02. ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		13

Диаграмма вариантов использования представлена на рисунке 6.

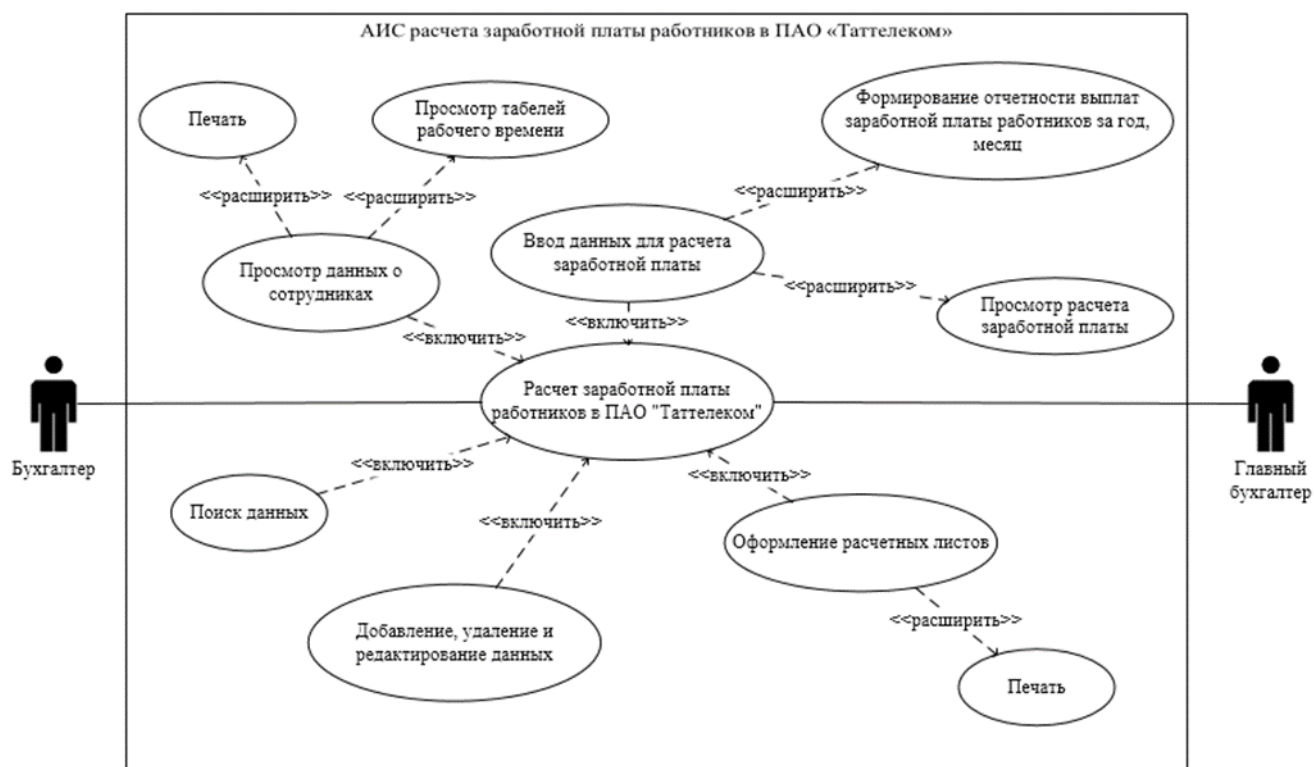


Рисунок 6 Диаграмма вариантов использования

Согласно диаграмме представлены основные возможности программы после авторизации как бухгалтера, так и главного бухгалтера. У главного бухгалтера есть дополнительные возможности: добавление, редактирование и удаление данных, а также возможность использования функционала бухгалтера.

1.2.2 Логическая модель информационной системы: Диаграмма классов

Логическая модель информационной системы (или диаграмма классов) служит для представления статической структуры модели системы в терминологии классов объектно-ориентированного программирования. Диаграмма классов может отражать, в частности, различные взаимосвязи между отдельными сущностями предметной области, такими как объекты и подсистемы, а также описывает их внутреннюю структуру и типы отношений.

Диаграмма классов деятельности бухгалтерии представлена на рисунке 8.

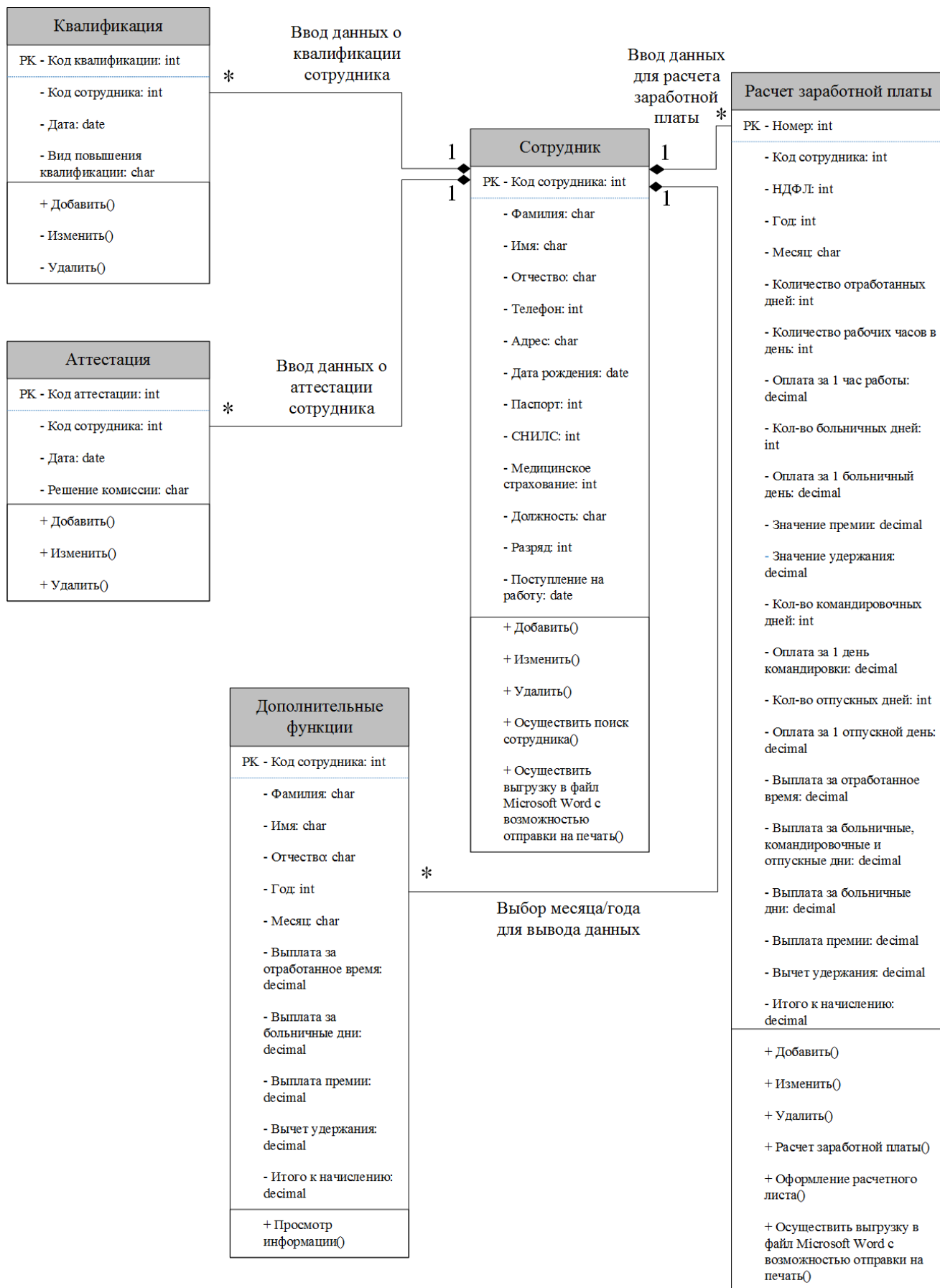


Рисунок 8 Диаграмма классов деятельности бухгалтерии

Согласно рисунку 8 в АИС деятельности бухгалтерии изображено 5 сущностей: Сотрудник, где хранится информация о сотрудниках; Квалификация и Аттестация, которые хранят в себе

учебные сведения; Расчет заработной платы, в которой присутствуют данные для расчета заработной платы работников в ПАО «Таттелеком»; Дополнительные функции, в которой имеются различные отчетные данные за месяц или год заработной выплаты работникам.

1.2.3 Модель на физическом уровне. Диаграмма компонентов

Диаграмма компонентов, в отличие от ранее рассмотренных диаграмм, описывает особенности физического представления системы. Диаграмма компонентов позволяет определить архитектуру разрабатываемой системы, установив зависимости между программными компонентами, в роли которых может выступать исходный, бинарный и исполняемый код. Во многих средах разработки модуль или компонент соответствует файлу. Пунктирные стрелки, соединяющие модули, показывают отношения взаимозависимости, аналогичные тем, которые имеют место при компиляции исходных текстов программ. Основными графическими элементами диаграммы компонентов являются компоненты, интерфейсы и зависимости между ними.

Диаграмма компонентов разрабатывается для следующих целей:

- визуализации общей структуры исходного кода программной системы;
- спецификации исполнимого варианта программной системы;
- обеспечения многократного использования отдельных фрагментов программного кода;
- представления концептуальной и физической схем баз данных.

В разработке диаграмм компонентов участвуют как системные аналитики и архитекторы, так и программисты. Диаграмма компонентов обеспечивает согласованный переход от логического представления к конкретной реализации проекта в форме программного кода. Одни компоненты могут существовать только на этапе компиляции программного кода, другие – на этапе его исполнения. Диаграмма компонентов отражает общие зависимости между компонентами, рассматривая последние в качестве классификаторов.

Диаграмма компонентов деятельности бухгалтерии представлена на рисунке 9.

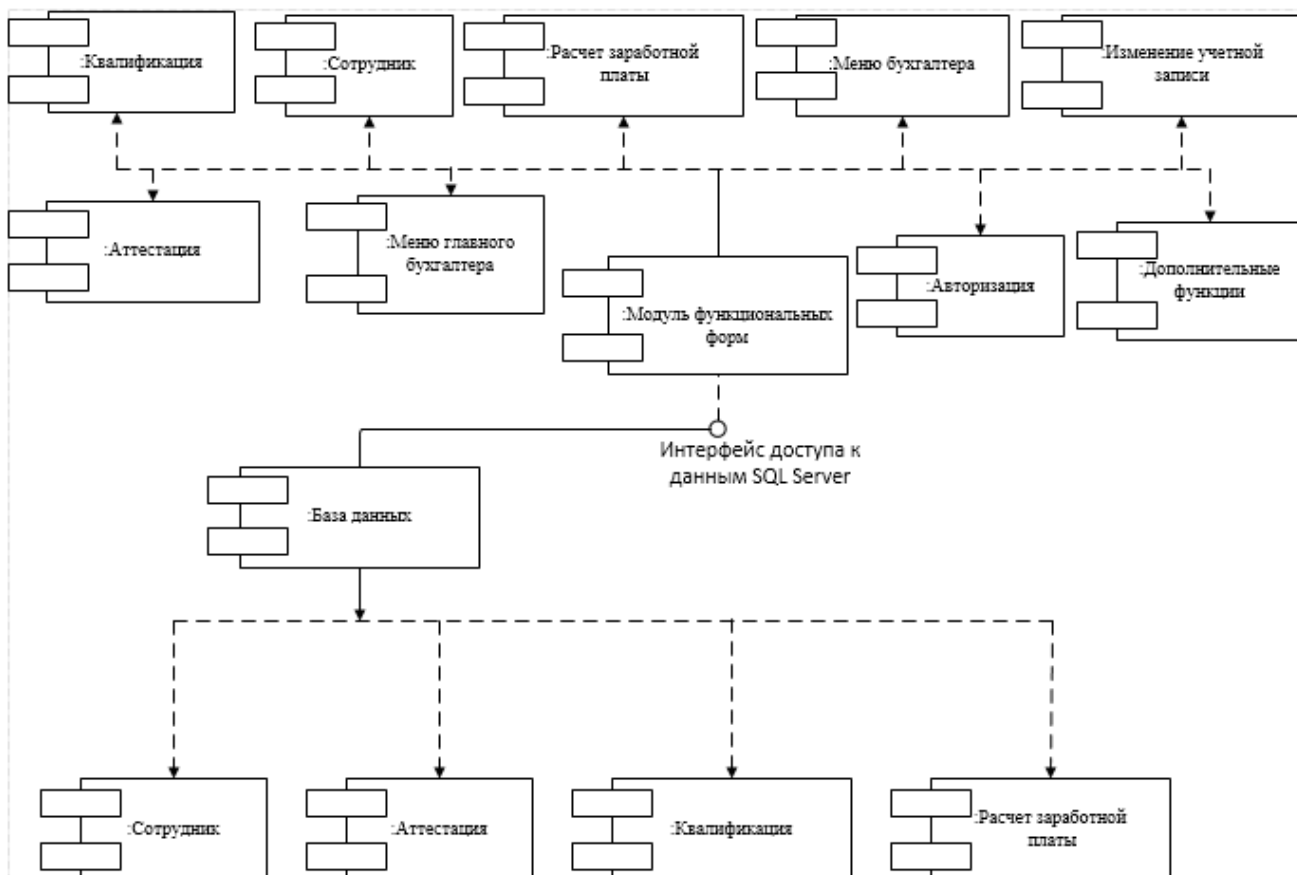


Рисунок 9 Диаграмма компонентов деятельности бухгалтерии

На диаграмме компонентов деятельности бухгалтерии показано разбиение программной системы на структурные компоненты и связи между ними, которые осуществляются через базу данных и модуль функциональных форм посредством интерфейса доступа к данным SQL Server. База данных включает в себя следующие сущности: Сотрудник, Аттестация, Квалификация, Расчет заработной платы, а модуль функциональных форм: Квалификация, Сотрудник, Аттестация, Авторизация, Расчет заработной платы, Меню бухгалтера, Меню главного бухгалтера, Изменение учетной записи, Дополнительные функции.