На рисунке 1 представлена диаграмма функционального моделирования в нотации (IDEF0) – «Как есть», которая отражает текущий процесс обработки заявки и добавления устройств в систему складского учёта. В данном случае, сотрудник организации создаёт много вспомогательных таблиц в офисных приложениях, добавляет в них необходимую информацию и тратит много временных ресурсов.

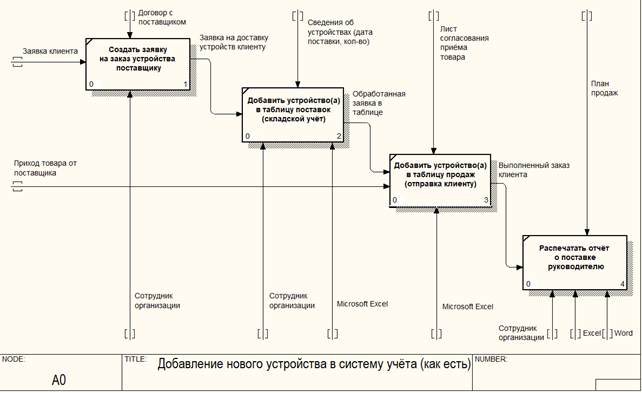


Рисунок 1 – Диаграмма функционального моделирования IDEF0 «Как есть»

Как видно из диаграммы, данные хранятся в отдельном файле, таблицы не нормализованы, поэтому определяется избыточность данных и, соответственно, нарушается их целостность. Предлагается автоматизировать процесс учёта путём создания базы данных и клиентского приложения. В разрабатываемой системе не нужно создавать много таблиц, спецификация и номенклатура создаётся отдельно для каждого добавляемого на склад устройства, доступ к которым можно получить прямо со своего рабочего места. Проектируемая система должна иметь возможность быстрого добавления и редактирования поступающих на склад комплектующих, формирования таблицы и статуса отправки товара клиенту, формировать отчёт о поступлении в текстовом виде, обеспечивать возможность печати и сохранения отчёта с соблюдением норм информационной безопасности.

На рисунке 2 представлена диаграмма по принципу функционального моделирования в нотации (IDEF0) – «Как надо».

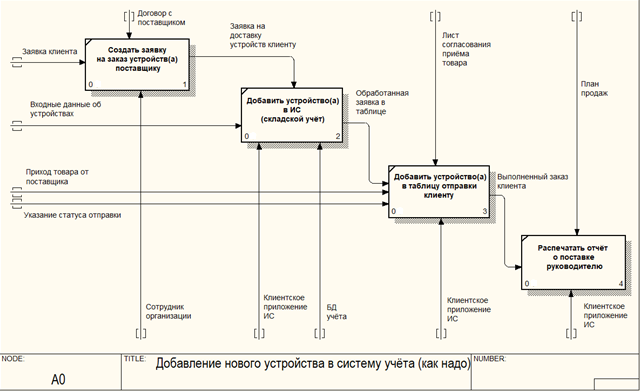


Рисунок 2 – Диаграмма функционального моделирования IDEF0 «Как надо»

Данная диаграмма отражает процесс учёта технических средств после внедрения информационной системы. Учёт осуществляется на основании поступления заявки от клиента, затем зарегистрированный в системе сотрудник связывается с поставщиком товара и при наличии такового оповещает клиента. Затем сотрудник организации получает товар на склад в установленный в договоре срок и вносит его в информационную систему учёта, на основе входных данных об устройстве (его характеристики). Сотрудник может добавить в систему спецификацию устройства или же руководство по использованию, которое впоследствии получит клиент.

После того, как устройство доставлено на склад, сотрудник организации связывается с заказчиком и оповещает его о доставке. Далее, на основании договора о предоставлении услуги по заказу оборудования, клиент получает устройство со склада. Сотрудник вносит данное устройство в таблицу отправки клиента, указывая статус отправки.

# Информационное обеспечение

# Данный раздел описывает поставленные перед разработкой задачи на языке моделирования. Приводятся UML – диаграммы проекта, а также описание основных этапов проектирования базы данных информационной системы.

# 2.1.1 Диаграммы бизнес - процессов

# Разрабатывая крупный программный продукт, возникает потребность в использовании универсального языка моделирования, поскольку при разработке системы в объектно – ориентированной парадигме, обязательно ставится цель декомпозиции – разбиения задач на более мелкие. Диаграммы UML собрали в себе лучшие решения инженеров, которые были проверены временем и показали свою пригодность при разработке профессиональных сложных программных систем. UML – диаграммы, как правило, представляют собой набор графических обозначений, для того, чтобы выразить дизайн будущего продукта и описать его функциональные особенности.

# В современной нотации UML – диаграммы разделяют на два типа – структурные диаграммы и диаграммы поведения. Далее будут рассмотрены диаграммы, которые использовались при проектировании информационной системы.

# В ходе осуществления проектирования были разработаны следующие диаграммы UML: диаграмма вариантов использования, диаграмма кооперации, диаграмма классов, диаграмма последовательности, диаграмма состояний, диаграмма компонентов.

# Рассмотрим данные диаграммы, определив их роль при проектировании информационной системы.

# На рисунке 3 представлена диаграмма вариантов использования, которая отражает отношения между пользователями системы («актёрами») и вариантами, предлагаемыми при использовании ИС. В качестве пользователей системы представлены: администратор базы данных, отвечающий за контроль по внесению изменений в информационную базу, восстановление БД в случае сбоя и работник организации, использующий клиентское приложение. Диаграмма вариантов использования отражает предъявленные к программному продукту функциональные требования, а также взаимосвязи между пользователями и самой программной системой.

# 

Рисунок 3 – Диаграмма вариантов использования ИС

# Далее рассмотрим диаграмму кооперации, которая отражает объекты и связи между ними в одном бизнес – процессе. Данная диаграмма представляет физическую реализацию процесса учёта, начиная от подачи заявки и заканчивая подтверждением и оповещением клиента об отправке. Диаграмма циклична, поскольку отражает реальный процесс. Основная цель – описать полный контекст взаимодействий, которые определяют выполнение задачи сотрудников по работе с клиентами. На рисунке 4 представлена диаграмма кооперации, отражающая реализацию обработки заявки от клиента.

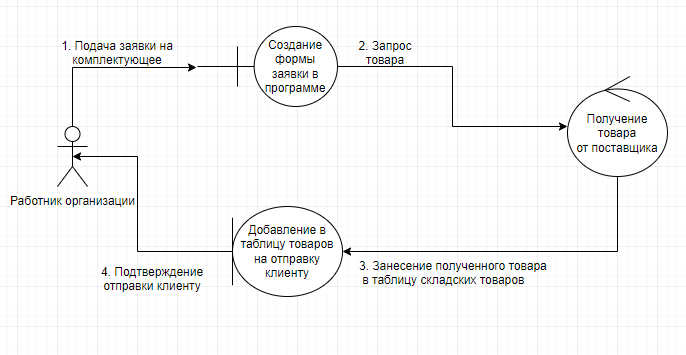


Рисунок 4 – Диаграмма кооперации

# На рисунке 5 изображена диаграмма классов, которая представляет собой прототип работы клиентского приложения. Были рассмотрены соединения экранных форм между собой, а также привязки сущностей из БД к клиентскому приложению. Диаграмма классов описывает принципы работы с редактированием устройств, а также формирование отчётов о доставке товаров на склад.

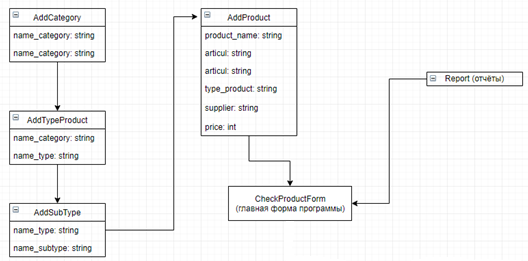


Рисунок 5 – Диаграмма классов

Диаграмма последовательности представляет собой жизненный цикл какого – либо процесса в программной системе. В данном случае рассматривается процесс авторизации и регистрации сотрудника в системе учёта. Данная диаграмма отражает все стадии авторизации пользователя, начиная с момента ввода данных и заканчивая подтверждением из БД о корректности введённых данных в систему.

Стрелками отображены сообщения к системе. В данном случае представлены: синхронное сообщение (отправитель передаёт ход управления получателю, который в соответствии с прецедентом организовывает действие или серию действий), ответное сообщение (ответ на синхронное сообщение о выполнении какого – либо действия, либо передачи прав).

Диаграмма последовательности представлена на рисунке 6.

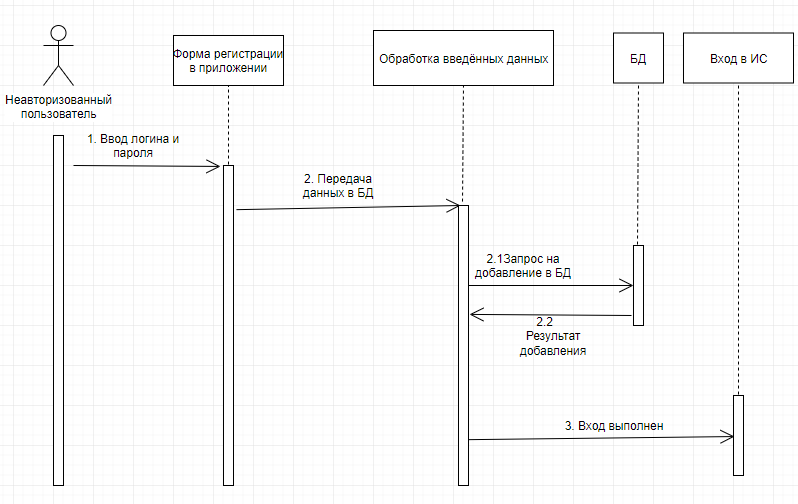


Рисунок 6 – Диаграмма последовательности регистрации сотрудника

Далее представлена диаграмма состояний, которая отражает состояние системы в процессе изменения различных факторов (событий). В данном случае рассматривается физическая реализация поставок оборудования на склад. Данная диаграмма представлена на рисунке 7.

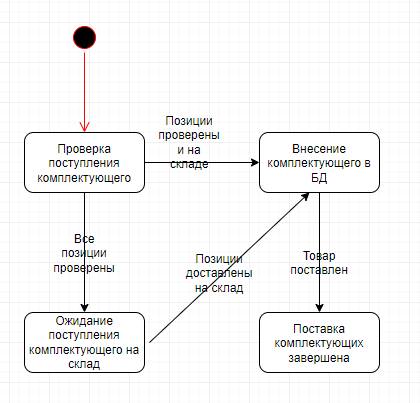


Рисунок 7 – Диаграмма состояний

Одной из важнейших UML – диаграмм является диаграмма компонентов, представленная на рисунке 8. Она используется для визуального отображения организации компонентов разрабатываемой системы и связей между ними. Преимуществом данной диаграммы является возможность представления декомпозиции задачи через сущности и порты, отображающие принципы взаимодействия компонентов ПО.

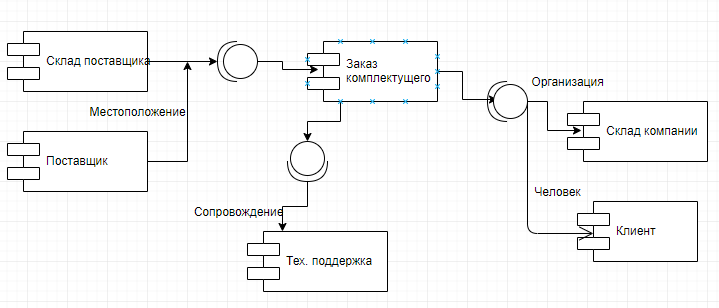


Рисунок 8 – Диаграмма компонентов