

Контрольная работа

Методологии функционального моделирования.

Цель работы: изучить методологию функционального моделирования IDEF0, получить практические навыки моделирования предметной области.

Порядок выполнения работы

1. Рассмотреть пример моделирования бизнес-процессов с применением стандарта IDEF0, а также пример ТЗ (Приложение А),
2. На основе описания предметной области «Учет и контроль аренды помещений» выполнить описание и построить функциональную диаграмму предметной области в нотации IDEF0 согласно варианту.
3. Разработать ТЗ согласно одному из предложенных в пункте 2 стандартов.
4. Оформить отчет по работе.

Требования к результатам выполнения контрольной работы

- Модель должна отражать бизнес-процессы предметной области
- Наличие в модели не менее 2 уровней: контекстная диаграмма, ее декомпозиция, декомпозиция одного из процессов первого уровня.
- Диаграмма декомпозиции должна включать не менее 3 функциональных блоков для нотации IDEF0.

При формировании **технического задания** обратить внимание на следующее:

- наличие пользовательских требований четко описывающий функционал разрабатываемого программного средства (не менее 20)
- наличие системных требований, включающих требования к структуре, программному интерфейсу, технологии разработки, общие требования к системе (надежность, модульность, безопасность и т.д.).

Отчет должен содержать:

- титульный лист;
- задание на контрольную работу согласно варианту;
- описание предметной области;
- таблицы 1.1-1.4;
- копии разработанных диаграмм (не менее трех: контекстная, ее декомпозиция, декомпозиция одного из процессов первого уровня);
- техническое задание.

1. ПРИМЕР МОДЕЛИРОВАНИЯ

1.1 Описание предметной области

В офисном здании в аренду сдаются помещения, следующую информацию о

которых необходимо хранить в базе данных: владелец помещения, расположение (схема), номер помещения, площадь, количество окон, степень изношенности, наличие кондиционера и др.

При обращении потенциального арендатора к администратору, осуществляется подбор нужного помещения согласно заданным критериям поиска, предварительный расчет суммы арендной платы в месяц. Стоимость аренды помещений оценивается исходя из коэффициентов, зависящих от многих факторов, в том числе, вышеуказанных характеристик помещения, также в итоговую стоимость аренды помещения могут входить иные коэффициенты, которые индивидуально оговариваются с арендатором. Если арендатора устраивает помещение и стоимость аренды, проходит встреча с владельцем, осмотр помещений с целью проверки заявленных характеристик помещения, обсуждение условий договора. Если обе стороны пришли к соглашению, то заключается договор аренды на определенный срок. Схема здания дополняется сведениями об арендаторах, расположенных в каждой комнате.

При заключении договора аренды администратор создает в базе нового пользователя, наделяя его оговоренными правами доступа к системе, а также фиксирует данные заключенного договора для своевременного и четкого контроля оплаты. По окончании срока аренды, договор расторгается либо продлевается на следующий срок. В случае расторжения договора, данному арендатору закрывается доступ к пользованию системой контроля аренды помещений и вносится статус о его благонадежности.

Каждый месяц арендатор вносит сведения об оплате, о чем сообщается арендодателю. В случае обнаружения задолженности, система автоматически осведомляет арендодателя и арендатора об ее наличии. В случае непогашения задолженности, договор может быть расторгнут ранее срока окончания.

Администратор регистрирует новых владельцев и арендаторов, вносит и редактирует данные о помещениях, договорах, может составлять статистики, а также отслеживать задолженности. Арендодатель может просматривать информацию о своих помещениях, заключенных договорах, платежах, составлять статистики.

Требуется автоматизировать учёт аренды помещений и контроль оплаты. Кроме сообщений об оплате и задолженности, необходимо получать периодические и срочные отчеты о договорах, оплате аренды и арендаторах с возможностью экспорта этих документов в текстовый редактор.

1.2 Анализ бизнес-процесса по методологии IDEF0

Процесс контроля и учета рассматривается с точки зрения Администратора.

Цель: заключение договоров и контроль арендной платы .

Таблица 1.1 – Условно-постоянная информация

№ пп	Информация	Исполнитель
1	Схема помещений	Администратор
2	Законодательство	Администратор

Таблица 1.2 – Входная информация

№ пп	Информация	Исполнитель
1	Сведения о владельцах	Внешний ресурс (арендодатель)
2	Поисковые запросы	Внешний ресурс (арендатор)
3	Данные об оплате аренды	Внешний ресурс (арендатор)

Таблица 1.3 – Выходная информация

№ пп	Информация	Исполнитель
1	Отчеты о договорах	Администратор
2	Отчеты об арендаторах	Администратор
3	Сообщения о задолженностях	Администратор
4	Отчеты об оплате	Администратор

Основной процесс контроля аренды помещений подразделяется на 3 процесса:

- учет занятости помещений;
- оформление договора;
- контроль исполнения договора.

Данные бизнес-процессов и типы интерфейсов приведены в таблице 1.4.

Таблица 1.4 – Данные бизнес- процессов

№ пп	Бизнес -процесс	Информация	Тип интерфейсной дуги
1	Ведение справочника помещений	Схема помещений	Управление
		Данные об арендаторе	Обратная связь по входу
		Схема помещений с арендаторами	Выходная
2	Оформление договора аренды	Схема помещений с арендаторами	Прямая связь по входу
		Запрос арендатора	Внешняя (граничная)
		Договор	Выходная
		Статус арендатора	Обратная связь по входу
		Отчеты о договорах	Выходная
		Отчеты об арендаторах	Выходная
3	Контроль	Договор аренды	Прямая связь по

№ пп	Бизнес -процесс	Информация	Тип интерфейсной дуги
	исполнения договора		управлению
		Сведения об оплате	Внешняя (граничная)
		Сообщения о задолженности	Выходная внешняя
		Статус арендатора	Выходная

1.2 Основы работы в *Erwin Process Modeler*. Создание контекстной диаграммы.

1. Загрузите *Erwin Process Modeler*. Создайте новую модель – Create model, имя модели: "Деятельность компании".

2. Заполните свойства модели – Propertie for New Models.

На вкладке *General* введите имя и инициалы автора, включите опцию **Apply CRUD/IRUN Restrictions** – Применить ограничения **CRUD/IRUN**.

Эти ограничения применяются для связывания модели процессов и модели данных. Данные не могут использоваться работами произвольно. Например, входные данные, представленные стрелкой входа, могут только преобразовываться в выход или потребляться. Рассматриваемые ограничения контролируют использование данных. Аббревиатура CRUD означает Create, Retrieve, Update, Delete и применяется для сущностей, а IRUN означает Insert, Retrieve, Update, Nullify (сделать неопределенным) и применяется для атрибутов сущностей

На вкладке **Numbering** задаются опции нумерации элементов модели

На вкладке **Display** определяется, что будет отображаться на диаграммах.

Параметры вкладки **Layout** – схема размещения отвечают за размещение объектов на диаграмме. Установите параметры согласно рисунку 1.1.

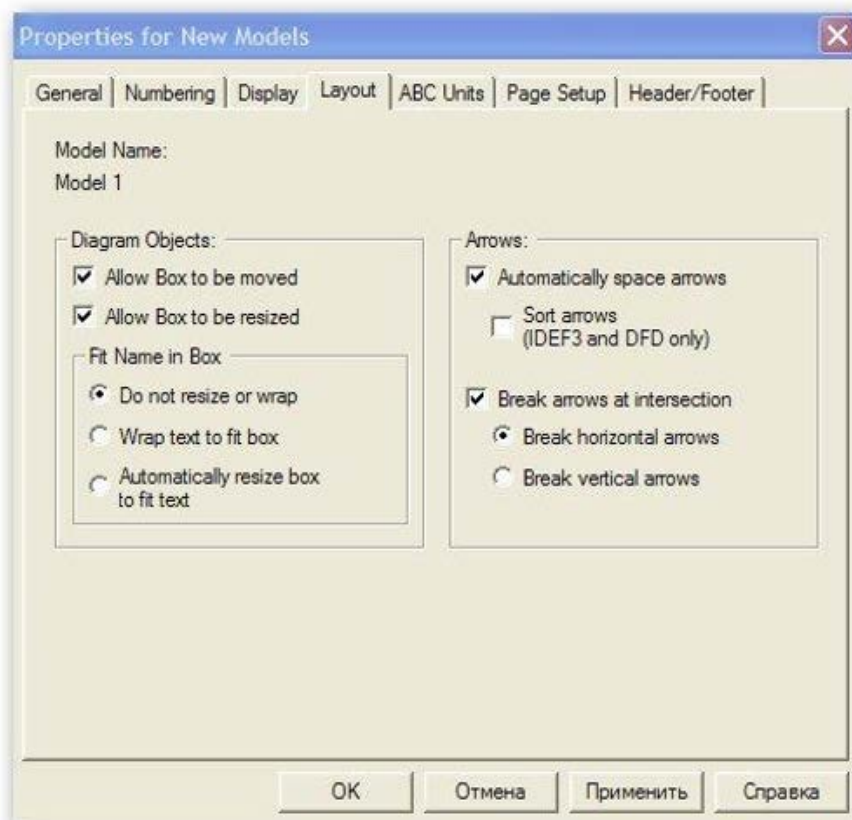


Рисунок 1.1 – Окно установки параметров размещения объектов на диаграмме

Вкладка **ABC Units** отвечает за содержание единиц стоимостного анализа. На вкладке **PageSetup** задаются опции просмотра и печати страницы.

На вкладке **Header/Footer** определяется пользовательский вид заголовка и нижнего колонтитула.

Сразу создается заготовка для контекстной диаграммы. Граничные рамки диаграммы называются каркасом. Каркас содержит заголовок (верхняя часть рамки) и подвал (нижняя часть).

3. Задайте имя заготовке контекстной диаграммы, выбрав свойства модели (меню *Model>Model Properties...*):

- Временные рамки Time Frame установите TO-BE
- На вкладке *Purpose* (Цель) установите цель моделирования Purpose: "Моделировать текущие бизнес–процессы компании" и точку зрения, с которой строится модель Viewpoint: "Аднимистоатор".
- На вкладке *Definition* (Определение) задайте определение модели Definition: "Учебная модель, описывающая деятельность компании" и границы (рамки) модели Scope: "Общее управление".

4. Дополните контекстную диаграмму стрелками. Чтобы нарисовать стрелку, необходимо выбрать соответствующую кнопку в палитре и первым кликом левой клавиши мыши обозначить начало стрелки, будь-то граница области диаграммы или граница функционального блока, а вторым кликом левой клавиши мыши – конец. Чтобы добавить имя стрелке, необходимо дважды кликнуть левой

клавишей мыши на стрелке или выбрать пункт Name в контекстном меню (После нажатия на кнопку Применить появляется окно полного диалога свойств стрелки).

Внимание! Никогда не подписывайте стрелки с помощью инструмента Text, так как в этом случае информация не попадет в словарь стрелок).

Полученная контекстная диаграмма показана на рисунке 1.2.

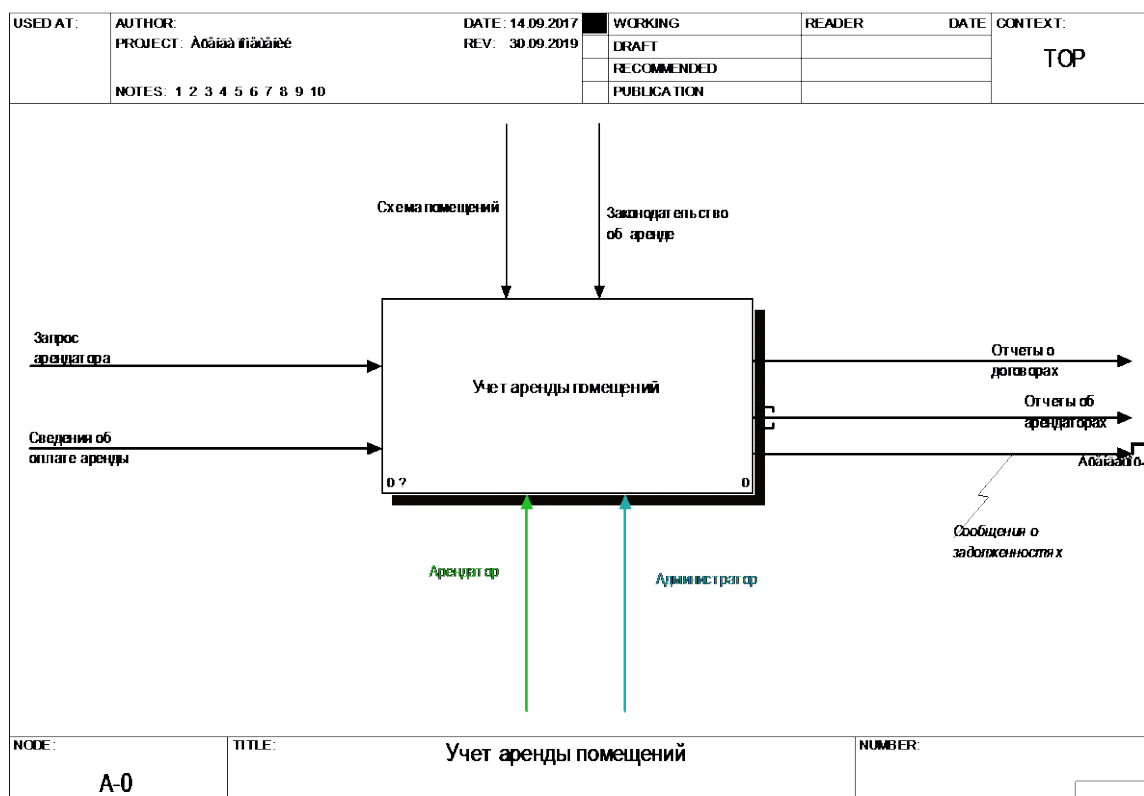


Рисунок 1.2 – Контекстная диаграмма

Для того чтобы добавить внешнюю ссылку(External Reference), нужно кликнуть правой клавишей мыши на конце стрелки у границы диаграммы и выбрать пункт External Reference. Открывается диалоговое окно, в котором можно ввести свое название внешней ссылки (оно автоматически будет добавлено в библиотеку), выбрать из уже добавленных или дать ей связь с соответствующей стрелкой. Свойства активностей и стрелок автоматически помещаются в соответствующие словари. Для просмотра словаря активностей выберите пункт меню Dictionary>Activity... Словарь стрелок можно вывести, выбрав пункт меню Dictionary>Arrow...

1. По модели можно создать отчет, содержащий выбранные пользователем свойства модели. В пункте меню Tools>Reports>ModelReport вызовите диалог Modelreport, в котором отмечают интересные свойства. Отчет может быть

предварительно просмотрен – Preview... , выведен на печать (Print...) или сохранен как текстовый файл.

1.2.2. Создание диаграмм декомпозиции

В результате анализа бизнес-процессов выделены три активности, составляющие процесс "Учет и контроль аренды помещений":

- учет занятости помещений;
- оформление договора;
- контроль исполнения договора.

1. Декомпозицию можно произвести двумя способами:

- выделить декомпозируемую контекстной диаграммы активность, кликнув на ней мышкой (или выделив блок в навигаторе модели на вкладке Activities), нажать кнопку Got o Child Diagram на панели инструментов.
- кликнуть правой клавишей мыши в навигаторе модели на вкладке Activities на блоке, который необходимо декомпонировать и из выпадающего списка выбрать пункт Decompose.

2. В диалоге *Activity Box Count*, выберите тип диаграммы – IDEF0 (декомпозицию можно провести в другой методологии) и число активностей на диаграмме декомпозиции –3.

Для связывания стрелок входа, управления и механизма необходимо перейти в режим редактирования стрелок (кнопка на панели инструментов), щелкнуть по наконечнику стрелки и щелкнуть по соответствующей стороне прямоугольника активности. Для связывания стрелки выхода необходимо в режиме редактирования стрелок щелкнуть по правой стороне активности, а затем – по стрелке. Щелкая по стрелке и соответствующей стороне блока активности, можно построить разветвление стрелки, например, стрелки правил. Для выхода из режима редактирования стрелок необходимо нажать кнопку указателя на панели инструментов.

3. Преобразуйте диаграмму по таблице 1.3 согласно рис. 1.3. Для рисования внутренней стрелки необходимо в режиме рисования стрелок щелкнуть на стороне блока активности, откуда выходит стрелка, затем по стороне блока активности, куда входит стрелка.

4. Задайте имена и свойства стрелок. Установив указатель мыши на имя стрелки (при этом выделяется и сама стрелка), можно перемещать имя, изменять размер поля имени. Указатель в виде молнии (Squiggle) включается с помощью контекстного меню. Можно использовать соответствующую кнопку на панели инструментов.

5. Для обратной связи «Сведения об аренде помещения» по входу между активностями "Оформление кредитного договора" и "Ведение справочника

- IEEE STD 830-1998
- ISO/IEC/ IEEE 29148-2011
- RUP

ГОСТ 34.602-89 Техническое задание на создание автоматизированной системы регламентирует структуру ТЗ на создание системы, в которую входят ПО, аппаратное обеспечение, люди, которые работают с ПО, и автоматизируемые процессы (см. приложение).

“ГОСТ 19.xxx Единая система программной документации (ЕСПД)” — это комплекс государственных стандартов, устанавливающих взаимосвязанные правила разработки, оформления и обращения программ (или ПО) и программной документации. Т.е. этот стандарт относится к разработке именно ПО.

IEEE STD 830-1998

Описывается содержание и качественные характеристики правильно составленной спецификации требований к программному обеспечению (SRS) и приводится несколько шаблонов SRS. Данная рекомендуемая методика имеет своей целью установление требований к разрабатываемому программному обеспечению, но также может применяться, чтобы помочь в выборе собственных и коммерческих программных изделий.

Согласно стандарту техническое задание должно включать следующие разделы:

1. Введение
 1. Назначение
 2. Область действия
 3. Определения, акронимы и сокращения
 4. Ссылки
 5. Краткий обзор
2. Общее описание
 1. Взаимодействие продукта (с другими продуктами и компонентами)
 2. Функции продукта (краткое описание)
 3. Характеристики пользователя
 4. Ограничения
 5. Допущения и зависимости
3. Детальные требования (могут быть организованы по разному, например)
 1. Требования к внешним интерфейсам
 - 1.1 Интерфейсы пользователя
 - 1.2. Интерфейсы аппаратного обеспечения
 - 1.3. Интерфейсы программного обеспечения

- 1.4. Интерфейсы взаимодействия
 2. Функциональные требования
 3. Требования к производительности
 4. Проектные ограничения (и ссылки на стандарты)
 5. Нефункциональные требования (надежность, доступность, безопасность и пр.)
 6. Другие требования
-
4. Приложения
 5. Алфавитный указатель

ISO/IEC/ IEEE 29148-2011

Стандарт IEEE 29148-2011 обеспечивает единую трактовку процессов и продуктов, используемых при разработке требований на протяжении всего жизненного цикла систем и программного обеспечения. Он приходит на смену стандартов IEEE 830-1998, IEEE 1233-1998, IEEE 1362-1998.

Данный стандарт содержит два шаблона спецификации требований:

- System requirements specification (SyRS)
- Software requirements specification (SRS)

System Requirements Specification (SyRS) определяет технические требования для выбранной системы и удобства взаимодействия предполагаемой системы и человека. Она определяет высокоуровневые требования к системе с точки зрения предметной области, а также информацию об общей цели системы, ее целевой среде и ограничениях, допущениях и нефункциональных требованиях. Она может включать в себя концептуальные модели, спроектированные для иллюстрации содержания системы, сценариев использования, основных сущностей предметной области, данных, информации и рабочих процессов. Из определения следует, что это аналог ТЗ, описанного в ГОСТ 34.

SyRS может содержать следующие разделы:

1. Введение
 1. Назначение системы
 2. Содержание системы (границы системы)
 3. Обзор системы
 1. Содержание системы
 2. Функции системы
 3. Характеристики пользователей
 4. Термины и определения
2. Ссылки
3. Системные требования

1. Функциональные требования
2. Требования к юзабилити
3. Требования к производительности
4. Интерфейс (взаимодействие) системы
5. Операции системы
6. Состояния системы
7. Физические характеристики
8. Условия окружения
9. Требования к безопасности
10. Управление информацией
11. Политики и правила
12. Требования к обслуживанию системы на протяжении ее жизненного цикла

13. Требования к упаковке, погрузке-разгрузке, доставке и транспортировке

4. Тестирование и проверка (список необходимых приемочных тестов, которые отражают зеркально раздел 3)

5. Приложения

1. Предположения и зависимости
2. Аббревиатуры и сокращений

SRS это спецификация требований для определенного программного изделия, программы или набора программ (продукт), которые выполняют определенные функции в конкретном окружении. Из определения следует, что это аналог ТЗ, описанного в ГОСТ 19, а по структуре очень напоминает SRS из стандарта IEEE 830.

SRS может содержать следующие разделы:

1. Введение
 1. Назначение
 2. Содержание (границы)
 3. Обзор продукта
 1. Взаимодействие продукта (с другими продуктами и компонентами)
 2. Функции продукта (краткое описание)
 3. Характеристики пользователей
 4. Ограничения
 4. Термины и определения
2. Ссылки
3. Детальные требования
 1. Требования к внешним интерфейсам
 2. Функции продукта
 3. Требования к юзабилити

4. Требования к производительности
5. Требования к логической структуре БД
6. Ограничения проектирования
7. Системные свойства ПО
8. Дополнительные требования
4. Тестирование и проверка (список необходимых приемочных тестов, которые отражают зеркально раздел 3)
5. Приложения
 1. Предположения и зависимости
 2. Аббревиатуры и сокращений

RUP

Структура SRS в RUP(Rational Unified Process) представляет собой документ, в котором необходимо описать артефакты, полученные в процессе специфицирования требований.

Традиционный шаблон SRS со структурированными функциональными требованиями по функциям Системы, максимально похож на 830 стандарт.

Упрощенный шаблон SRS со структурированными функциональными требованиями в виде вариантов использования (usecases):

- 1 Введение.
 1. Цель.
 2. Краткая сводка возможностей.
 3. Определения, акронимы и сокращения.
 4. Ссылки.
 5. Краткое содержание.
2. Обзор системы
 1. Обзор вариантов использования.
 2. Предположения и зависимости.
3. Детальные требований
 1. Описание вариантов использования.
 2. Дополнительные требования.
 3. Другие функциональные требования.
 4. Нефункциональные требования.
4. Вспомогательная информация.

3 ВАРИАНТЫ ЗАДАНИЙ

1. Опишите процесс организации работы с нарушителями правил дорожного движения с точки зрения работника ГАИ.
Разработать программный модуль «Учет нарушений правил дорожного движения». Для каждой автомашины (и ее владельца) в базе хранится список нарушений. Для

каждого нарушения фиксируется дата, время, вид нарушения и размер штрафа, также сведения об оплате штрафа. При оплате всех штрафов авто не удаляется из базы.

2. Опишите процесс работы автомагазина с точки зрения его служащего.

Разработать программный модуль «Автомагазина», предназначенный для использования работниками магазина. В базе содержатся сведения об автомобилях (марка, объем двигателя, дата выпуска и др.). При поступлении заявки на покупку производится поиск подходящего варианта. Если такого нет, клиент заносится в клиентскую базу и оповещается, когда вариант появляется. Покупка фиксируется в базе данных.

3. Опишите процесс работы АТС с точки зрения ее служащего.

Разработать программный модуль «Картотека абонентов АТС». Картотека содержит сведения о телефонах и их владельцах. Фиксирует задолженности по оплате (абонентской и повременной). Считается, что повременная оплата местных телефонных разговоров уже введена.

4. Опишите процесс организации работы автостанции с точки зрения ее служащего.

Разработать программный модуль «Автокасса», содержащий сведения о наличии свободных мест на автобусные маршруты. В базе должны содержаться сведения о номере рейса, маршруте, водителе, типе автобуса, дате и времени отправления, а также стоимости билетов. При поступлении заявки на билеты программа производит поиск подходящего рейса, наличие мест – возможность выбора.

5. Опишите процесс работы книжного магазина с точки зрения его служащего.

Разработать программный модуль «Книжный магазин», содержащий сведения о книгах (автор, название, издательство, год издания, цена). Покупатель оформляет заявку на нужные ему книги, если таковых нет, он заносится в базу и оповещается, когда нужные книги поступают в магазин.

6. Опишите процесс работы автостоянки с точки зрения ее служащего.

Разработать программный модуль «Автостоянка». В программе содержится информация о марке автомобиля, его владельце, дате и времени въезда, стоимости стоянки, скидках, задолженности по оплате и др.

7. Опишите процесс организации работы гостиницы с точки зрения администратора.

Разработать программный модуль «Гостиница», содержащий сведения о наличии свободных мест и о проживающих в гостинице. Программный модуль предназначен для бронирования мест в гостинице и оформления проживающих.

8. Опишите процесс организации работы детективного агентства с точки зрения ее работников.

Разработать программный модуль «Детективное агентство», содержащий сведения о клиентах агентства и об оказанных услугах. Программный модуль предназначен для учета средств за оказанные услуги и исполнения заказа.

9. Опишите процесс работы музея с точки зрения его служащего.

Разработать программный модуль «Музей», предназначенный для использования работниками музея. В базе содержатся сведения об экспонатах музея и вносятся данные при поступлении новых экземпляров. При выполнении инвентаризации данные заносятся в базу, проводится сверка и выдаются отчеты по учету экспонатов в музее., фиксируются данные о перемещении (реставрация, архив, выставка).

10. Опишите процесс работы автосервиса с точки зрения его администратора.

Разработать программный модуль «Автосервис», предназначенный для использования сотрудниками автосервиса. В базе содержатся сведения о ремонтируемых марках авто, выполняемых работах, поставщиках запчастей. При обращении клиента оформляется заказ, осуществляется проверка наличия клиента в базе, данные заказа заносятся в базу, и выдаются отчеты по заказам.

11. Опишите процесс работы локомотивного депо с точки зрения его сотрудника.

Разработать программный модуль «Депо», предназначенный для использования его сотрудниками. В базе содержатся сведения о локомотивных бригадах, локомотивах, их обслуживании и ремонте, видах ремонта и обслуживания. Техническое обслуживание и текущий ремонт выполняются согласно плану в зависимости от пробега локомотива. Внеплановый ремонт выполняется при необходимости. Необходимо предусмотреть формирование планов ремонта, контроль их выполнения, а также учет внеплановых ремонтов.

12. Опишите процесс работы отдела снабжения предприятия с точки зрения его сотрудника.

Разработать программный модуль «Снабжение», предназначенный для использования его сотрудниками. В базе хранятся сведения о поставщиках, категории товара, товарах (материалах), сотрудниках. При поступлении заявки от подразделения предприятия назначается сотрудник, ответственный за выполнение заявки, который осуществляет поиск товара, формирует заказ для поставщика, осуществляется проверка наличия поставщика в базе, данные заказа заносятся в базу, фиксируется выполнения заказа и заявки, формируются отчеты по заявкам и заказам.

13. Опишите процесс работы отдела продаж с точки зрения его сотрудника.

Разработать программный модуль «Продажи», предназначенный для использования его сотрудниками. В базе хранятся сведения категории товара, товарах, их наличии, количестве и цене. При поступлении заявки от клиентов осуществляется проверка наличия сведения о клиенте в базе, фиксируются данные

заказа , если товар отсутствует, формируется заявка на поставку, формируются отчеты по продажам.

14. Опишите процесс работы проектного отдела строительной организации с точки зрения его руководителя.

Разработать программный модуль «Проекты», предназначенный для использования его руководителем. В базе хранятся сведения конструкторов, проектных работах, заказчиках. При поступлении заказа от организации заключается договор, где фиксируется вид проектных работ, даты проекта, назначаются исполнители: ведущий конструктор и конструкторы, по окончании проекта фиксируется дата сдачи акта приемки работ, формируются отчеты по проектам.

15. Опишите процесс работы туристической компании с точки зрения ее руководителя.

Разработать программный модуль «Туристическая компания», предназначенный для использования его руководителем. В базе хранятся сведения о менеджерах, турах по категориям, направлениям, каждый менеджер специализируется на одном или нескольких направлениях. При обращении клиента осуществляется проверка наличия сведений о нем в базе, заключается договор, данные договора заносятся в базу, формируются отчеты по турам, сформированным группам.

16. Опишите процесс работы отдела бронирования авиабилетов с точки зрения его сотрудников.

Разработать программный модуль «Бронирование авиабилетов», предназначенный для использования его сотрудниками. В базе хранятся сведения об авиакомпаниях, рейсах, расписании полетов. При обращении клиента осуществляется проверка наличия сведений о нем в базе, осуществляется поиск рейса, проверка наличия билетов на требуемую дату, фиксируются данные брони, формируются отчеты по наличию билетов, сводные отчеты по датам, авиакомпаниям.

17. Опишите процесс работы ломбарда с точки зрения его сотрудников.

Разработать программный модуль «Ломбард», предназначенный для использования его сотрудниками. При обращении клиента, заключается договор, в котором указываются сведения о клиенте, характеристики ценности, включая ее стоимость, и срок договора. Программный модуль предназначен для учета хранимых ценностей и контроля возврата денег.

18. Опишите процесс работы фитнес-клуба с точки зрения его администратора.

Разработать программный модуль «Фитнес», предназначенный для использования его администратором. В базе хранятся сведения об инструкторах, направлениях, каждый инструктор специализируется на одном или нескольких направлениях, расписании групп. При обращении клиента осуществляется выбор группы, проверка свободных мест, заключается договор, данные договора заносятся в базу, формируются отчеты по турам, сформированным группам, осуществляется контроль оплаты.

19. Опишите процесс работы концертного зала с точки зрения его администратора.

Разработать программный модуль «Концерты», предназначенный для использования его администратором. При заключении договора с представителем артиста (артистов) фиксируется количество и даты выступлений, райдер артиста, сумма вознаграждения, порядок оплаты. После подписания договора администратор ведёт учет затрат, для каждого договора фиксируется вид платежного документа, назначение (реклама, гостиница и т.п.), сумма оплаты, а также организовывает продажу билетов, количество проданных билетов также вносится в базу. После выступлений для каждого договора формируется отчет о затратах и расчет прибыли.

20. Опишите процесс работы ресторана с точки зрения его администратора.

Разработать программный модуль «Ресторан», предназначенный для использования его администратором. Администратор ведет учет заказов столиков, а также возможных банкетов (свадеб, юбилеев и т.п.). При заказе столика осуществляется проверка свободных мест на определенную дату и при их наличии столик бронируется. При заказе банкета фиксируются количество человек и предварительная стоимость, после согласования меню и прочих услуг, вносится окончательная сумма, сумма предоплаты и ставится отметка о согласовании.

Вариант выбирается по таблице 5: строка – первая буква фамилии, столбец первая буква имени, на пересечении – номер варианта.

Например, Остап Бендер – 3 вариант.

Таблица 5 – Варианты

Имя фамилия	А, Я	Б, П	В, Ф	Г, Р	Д, Т	Е, Э	Ж, Ш	З, С	И, Щ	К, Х	Л, О	М, Ц	Н, Ч	У, Ю
А, Я	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Б, П	3	5	7	9	11	13	15	17	19	1	3	5	7	9
В, Ф	4	6	8	10	12	14	16	18	20	2	4	6	8	10
Г, Р	19	20	1	2	3	4	5	6	7	12	11	10	9	8
Д, Т	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Е, Э	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Ж, Ш	7	8	9	10	11	12	13	14	15	16	17	18	19	20
З, С	8	9	10	11	12	13	14	15	16	17	18	19	20	1
И, Щ	17	19	11	3	5	7	9	11	13	15	17	19	1	20
К, Х	9	11	15	17	19	2	4	6	8	10	12	13	15	17
Л, О	11	12	13	14	15	16	17	18	19	20	1	2	3	4
М, Ц	10	12	14	18	20	2	5	4	6	11	13	15	17	19
Н, Ч	13	14	15	16	17	18	19	20	1	2	3	4	5	6
У, Ю	14	16	18	1	3	5	7	9	11	13	15	17	19	21

ПРИМЕР ТЕХНИЧЕСКОГО ЗАДАНИЯ В СООТВЕТСТВИИ С ГОСТ 34.602-89

1. Общие сведения

1.1. Наименование

Программное средство «Учет и контроль аренды помещений».

1.2. Область применения

Данная разработка предназначена для применения в отделе аренды компании ООО «»

1.3. Основание для разработки

Программа разрабатывается в рамках контрольной работы по технологии разработки программного обеспечения.

2. Назначение, цели и задачи ПО

2.1. Назначение разработки

Программа предназначена для использования сотрудниками компании ООО « » с целью автоматизации процессов учета помещений, сдаваемых в аренду, и контроля оплаты.

2.2. Задачи ПО

Программное средство должно обеспечить хранение в базе данных информации о характеристиках помещений, поиск свободного помещения по заданным критериям, оформление и хранение сведений о договорах на аренду, контроль оплаты за аренду, подготовку периодических и срочных отчетов.

3. Технические требования к программе или программному изделию

3.1. Требования к функциональным характеристикам

3.1.1. Функциональные требования

Программа должна обеспечивать возможность выполнения следующих функций:

- ведение справочников помещений;
- проверка наличия информации об арендаторе в БД;
- ввод и корректировка информации об арендаторах;
- ввод, корректировка и удаление информации о договорах;
- поиск сведений о помещении по запросу арендатора;
- ввод данных об оплате;
- контроль оплаты;
- автоматическую отправку сообщений о задолженности арендатору и арендодателю;
- формирование отчетов о помещениях, арендаторах;
- формирование отчетов по договорам.

3.2. Исходные данные

- Запросы арендатора.
- Схема помещений с их характеристиками.

3.3. Требования к безопасности

В разрабатываемой системе необходимо предусмотреть следующие меры защиты:

- контроль вводимой информации;
- разграничение прав доступа;

- защиту от несанкционированного доступа посредством паролей;
- возможность резервного копирования;
- автоматического сохранения изменений после завершения транзакций.

3.4. Требования к надежности

Время восстановления после отказа, вызванного сбоем электропитания технических средств (иными внешними факторами), не фатальным сбоем операционной системы, не должно превышать времени, необходимого на перезагрузку операционной системы и запуск программы.

Время восстановления после отказа, вызванного неисправностью технических средств, фатальным сбоем (крахом) операционной системы, не должно превышать времени, требуемого на устранение неисправностей технических средств и переустановки программных средств.

3.5. Условия эксплуатации

Минимальное количество персонала, требуемого для работы программы, должно составлять не менее 2 штатных единиц - системный программист и конечный пользователь программы –администратор здания.

Системный программист должен иметь минимум среднее техническое образование. В перечень задач, выполняемых системным программистом, должны входить:

- задача поддержания работоспособности технических средств;
- задачи установки (инсталляции) и поддержания работоспособности системных программных средств - операционной системы;
- задача установки (инсталляции) программы.

Конечный пользователь программы (менеджер по продажам) должен обладать практическими навыками работы с графическим пользовательским интерфейсом операционной системы.

3.6. Требования к составу и параметрам технических средств

В состав технических средств должен входить IBM-совместимый персональный компьютер (ПЭВМ), включающий в себя:

- и так далее...

3.7. Требования к информационной и программной совместимости

Системные программные средства, используемые программой, должны быть представлены локализованной версией операционной системы Windows XP (Windows Vista, Windows 7).

3.8. Специальные требования

Программа должна быть снабжена графическим интерфейсом.

4. Стадии и этапы разработки

4.1. Стадии разработки

Разработка должна быть проведена в три стадии:

- разработка технического задания;
- рабочее проектирование;
- внедрение.

4.2. Этапы разработки

На рисунке представлена диаграмма Ганта для проекта разработки программного модуля.

Файлы проекта и планы работ для участников прилагаются.¹

5. Порядок контроля и приемки

После проведения испытаний в полном объеме, на основании «Протокола испытаний» утверждают «Свидетельство о приемке», после чего программный продукт считается принятым.

¹ Выполнить лабораторную работу «Управление программным проектом в MSProject»