ГОУ ВПО – Донецкий Национальный Университет

Физико-Технический факультет

Лабораторная работа по дисциплине

«ТЕХНОЛОГИЯ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ»

Кафедра Компьютерных Технологий

Подготовила студентка группы ИВТ-6

Печёнкина Галина

Донецк 2018



**ЛАБОРАТОРНАЯ РАБОТА №1**

Курс «Технологии разработки программного обеспечения»

**Тема:** Составление спецификаций требований заказчика программного продукта.

**Цель:** Научиться формализовать требования заказчика программного продукта и оформлятьтехническое задание на создание программного продукта.

**Примечание к лабораторным работам**.

* рамках лабораторных работ №1-№5 студент занимается разработкой одного программного продукта. В роли заказчика продукта выступает проработчик курса «Технологии создания программного продукта» на кафедре компьютерных технологий. Каждому студенту необходимо разработать простейшую информационную систему по определенной предметной области. Предметную область студент может либо предложить самостоятельно (например, совпадающую с тематикой его курсовой работы), либо получить по варианту (приложение 1.1). Предметные области у разных студентов не могут повторяться.

**Задание:**

1. Провести интервью с заказчиком, на основе которого сформулировать цель и задачи проекта по созданию программного продукта.
2. На основе интервью с заказчиком определить пользователей продукта.
3. На основе интервью с заказчиком определить нефункциональные требования к продукту.
4. Провести дополнительно самостоятельный анализ предметной области.
5. Сформировать спецификации вариантов использования продукта.
6. Представить модель объекта автоматизации (информационной системы) в виде блок-схем и/или простого текста (примеры – приложение 1.2).
7. Оформить техническое задание на создание программного продукта. В качестве стандарта использовать ГОСТ 34.602-89 или IEEE Std.830-1998.
8. Оформить отчет, включающий ответы на контрольные вопросы, составленное техническое задание.

**Контрольные вопросы:**

1. Кратко опишите основные этапы разработки программного продукта.
2. Что представляют собой артефакты программного продукта?
3. Опишите основные стратегии разработки ПО.
4. Кратко охарактеризуйте системы принципов разработки ПО (PSP, TSP, CMM).
5. Состав персонала разработки программного продукта.
6. Укажите основные метрики качества программного продукта.
7. В чем заключается процесс инспектирования проекта? Укажите принципы инспектирования.
8. Приведите примеры функциональных и нефункциональных требований к программному продукту.
9. Что представляют собой варианты использования программного продукта?
10. Приведите схему процесса анализа С-требований.
11. Приведите схему процесса анализа D-требований.

ПОСТАНОВКА ЗАДАЧИ

Объектом автоматизации является отдел обслуживания покупателей в магазине музыкальных инструментов, предоставляющий покупателям выбрать искомый инструмент за определенную цену, а также узнать о всех характеристиках данного товара. Задача ПП заключается в упрощении работы сотрудников с покупателями, а также в приросте покупателей и развитии магазина. Сотрудник магазина может задать параметры поиска: дата продажи, наименование продажи, количество проданных товаров за определенных срок и т.д. Получить информацию о существующих продажах, наличии инструментов на складе, данных покупателей и сотрудников. Функциями отдела продаж является:

* предоставление информации о существующих товарах;
* ведение учета товаров;
* осуществление продаж;
* добавление новых покупателей в каталог покупателей.

Структура рассматриваемого отдела обслуживания покупателей представлена на рисунке 1, каждый из которых обеспечивает выполнение возложенных на него задач.



Рисунок 1 – Схема организационной структуры отдела обслуживания покупателей магазина музыкальных инструментов

Администратор представляет собой специалиста, работающего с БД. Бухгалтер-кассир занимается продажей товара и подсчетом доходов. Менеджер по продажам управляет всеми процессами в отделе, планирует его работу и консультирует покупателей. Кладовщик оформляет поставку товаров и регулирует работу на складе.

Рассмотрев схему организационной структуры отдела обслуживания покупателей магазина музыкальных инструментов, приведенную на рисунке 1, можно сделать вывод о линейно-функциональной структуре управления данной организацией. При такой структуре управления общий объем работы четко распределяется между сотрудниками, и каждый специалист несет ответственность за свою часть выполняемой работы.

ПС должна вести нижеуказанную информацию по следующим объектам:

|  |  |  |  |
| --- | --- | --- | --- |
| Покупатели | Сотрудники | Инструменты | Продажи |
| ID покупателя | ID сотрудника | ID инструмента | ID продажи |
| ФИО покупателя | ФИО сотрудника | Наименование | Наименование |
| Телефон | Должность | Категория | Категория |
| Адрес | Телефон | Модель | Модель |
|  | Адрес | Фирма | Фирма |
|  |  | Страна | Стоимость |
|  |  | Год | ФИО покупателя |
|  |  | Стоимость | ФИО сотрудника |
|  |  | Количество на складе | Количество |
|  |  | Фото | Дата |
|  |  |  | Фото |

Объект исследования – автоматизированная информационная система учета товаров в магазине музыкальных инструментов.

Функциями ПП является: редактирование всей информации в программе; просматривание записей базы данных (в том числе и при помощи поиска) и их редактирование; добавление новых записей, а также их удаление.

Область применения – программа предназначена для использования персоналом для эффективного управления ассортиментом и возможности увеличения покупательской способности клиентов за счет повышения качества обслуживания. Сотрудник может задать параметры поиска и по заданным параметрам получать информацию о наличии продажи, наличии инструмента на складе, его цену, марку, фирму и год изготовления, список покупателей и данные по нему. Сотрудник имеет возможность вносить коррективы в базу данных (добавлять, редактировать, удалять, менять данные о продажах, покупателях, инструментах, сотрудниках).

1. **Требования к программе или программному изделию:**
   1. **Требования к функциональным характеристикам:**

Интерфейс должен предполагать стандартную системную цветовую палитру и разрабатываться под разрешение экрана 800х600. Оконные элементы управления и размер шрифта должны быть несколько больше стандартных размеров. Окна должны обладать системным меню с кнопкой закрытия.

ГЛАВНОЕ ОКНО, в котором должны быть вкладки «Продажи», «Сотрудники», «Покупатели», «Инструменты». В каждой вкладке отображается информация по выбранным инструментам, сотрудникам, продажам и покупателям, соответственно. На каждой вкладке должны быть кнопки «Изменить», «Добавить», «Удалить» для редактирования данных. В главном окне должно быть также меню с пунктом «Настройки».

Программа должна обеспечить возможность выполнения следующих функций:

* редактирование информации базы данных;
* просматривание записей базы данных;
* обработки запросов;
* добавление и удаление записей.

Исходные данные: таблицы записей информации о Покупателях (ID покупателя, Фамилия покупателя, Имя покупателя, Телефон, Адрес), Сотрудниках (ID сотрудника, ФИО сотрудника, Должность, Телефон, Адрес), Инструментах (ID инструмента, Наименование, Категория, Модель, Фирма, Страна, Год, Стоимость за единицу, Количество на складе, Фото) и Продажах (ID продажи, Наименование, Категория, Фирма, Модель, Стоимость, Имя покупателя, Фамилия покупателя, Имя сотрудника, Фамилия сотрудника, Количество, Дата, Фото).

Организация входных и выходных данных. Входные данные поступают с клавиатуры. Выходные данные отображаются на экране и при необходимости выводятся на печать.

* 1. **Требования к надежности:**

Предусмотреть контроль вводимой информации. Предусмотреть блокировку некорректных действий пользователя при работе с системой.

* 1. **Требования к составу и параметрам технических средств:**

Система должна работать на IBM-совместимых персональных компьютерах.

Минимальная конфигурация:

1. тип процессора: Pentium и выше;
2. объем оперативного запоминающего устройства: 512 Мб и более;
3. объем свободного места на жестком: диске 40 Мб.

Рекомендуемая конфигурация:

1. тип процессора: Pentium 4;
2. объем оперативного запоминающего устройства: 1024 Мб;
3. объем свободного места на жестком Диске: 60 Мб.
   1. **Требования к программной совместимости:**

Программа должна работать под управлением семейства операционных систем Win32 (Windows 95/98/2000/МЕ/ХР/7/8/10 и т. п.), но проект программного продукта должен иметь возможность поддержки в будущем кроссплатформенности.

* 1. **Требования к программной документации:**

1. Разрабатываемые программные модули должны быть самодокументированы, т.е. тексты программ должны содержать все необходимые комментарии.
2. Разрабатываемая программа должна включать справочную информацию о работе программы, описание работы с ней.
   1. **Требования к пользователям продукта:**

Администратор:

* ­­­Понимание принципов функционирования систем управления БД.

Директор:

* Владение компьютером на уровне пользователя ОС Windows.

Менеджер:

* Владение компьютером на уровне пользователя ОС Windows.
  1. **Ограничения:**
* Должна использоваться кодировка UTF-8.
* Продукт будет поддерживать только русский язык пользовательского интерфейса.

**Варианты использования.**

1. ***Вариант использования «Авторизация»:***
   1. Пользователь открывает файл программы.
   2. Система отображает окно «Авторизация» с полями для ввода логина и пароля, а также кнопкой «Войти».
   3. Пользователь в появившемся окне заполняет поля в соответствии со своей должностью.
   4. Пользователь нажимает кнопку «Войти».
   5. Система закрывает окно «Авторизация» и отрывает окно «Меню».
2. ***Вариант использования «Изменение данных для входа в программу»:***
   1. Администратор в окне «Сотрудники» нажимает на кнопку «Редактировать».
   2. Система отображает окно «Редактировать» с информацией о сотруднике.
   3. Администратор при помощи системного меню выбирает необходимого сотрудника.
   4. Система отображает информацию о сотруднике, включая логин и пароль.
   5. Администратор заполняет поля авторизации, добавляя для сотрудника новый логин и пароль.
   6. Администратор нажимает кнопку «Сохранить».
   7. Система сохраняет новые данные и закрывает окно «Редактировать».
3. ***Вариант использования «Добавить сотрудника»:***
   1. *Администратор* в окне с информацией о сотрудниках нажимает кнопку «Добавить сотрудника».
   2. *Система* отображает окно «Добавление сотрудника» с полями для ввода ID сотрудника, ФИО, адреса, телефона, должности, а также логина и пароля и с наличием навигатора, соответственно.
   3. *Администратор* в появившемся окне заполняет все поля и нажимает кнопку «Добавить».
   4. *Система* закрывает окно «Добавить сотрудника» и добавляет нового сотрудника в список существующих.
4. ***Вариант использования «Добавление покупателя»:***
   1. Менеджер в окне «Продажи» нажимает на кнопку «Добавление продажи»;
   2. Система отображает окно «Добавление продажи» с необходимой информацией;
   3. Менеджер нажимает на кнопку «Добавить» возле информации о покупателе;
   4. Система отображает дополнительное окно «Добавить покупателя» с полями для ввода ID покупателя, фамилии, имени, адреса, контактного телефона и панелью навигации, соответственно;
   5. Менеджер заполняет все поля;
   6. Менеджер нажимает кнопку «Добавить» на панели навигации;
   7. Система добавляет покупателя в форму «Покупатели» и закрывает окно «Добавление покупателя»;
   8. Система вписывает покупателя в текущую продажу в окне «Добавление продажи».
5. ***Вариант использования «Добавление продажи»:***
   1. Менеджер в окне «Продажи» нажимает на кнопку «Добавление продажи»;
   2. Система отображает окно «Добавление продажи» со следующими данными: ID продажи, дата продажи, список товаров, ID инструмента, ID покупателя, имя покупателя, фамилия покупателя, ID сотрудника, имя сотрудника, фамилия сотрудника, количество покупаемых товаров, общая стоимость, фото покупаемого товара и панель навигации;
   3. Менеджер выбирает покупаемый товар из общего списка товаров;
   4. Система отображает фото выбранного товара, автоматически заполняя поле ID инструмента;
   5. Менеджер выбирает фамилию покупателя из списка покупателей или нажимает на кнопку «Добавить покупателя»;
   6. Система автоматически добавляет имя текущего покупателя или переходит в окно «Добавление покупателя»;
   7. Менеджер выбирает из списка сотрудников свою фамилию;
   8. Система автоматически отображает имя сотрудника и его ID;
   9. Менеджер заполняет поле «Количество»;
   10. Система подсчитывает стоимость товара за заданное количество единиц и подсчитывает общую стоимость с учётом скидки в том случае, если покупатель до этого приобретал уже товары;
   11. Менеджер нажимает на кнопку «Добавить продажу» на панели навигации;
   12. Система добавляет новую продажу в окно «Продажи».
6. ***Вариант использования «Добавление товара»:***
   1. Администратор в окне «Каталог инструментов» нажимает на кнопку «Добавить»;
   2. Система отображает окно «Добавление товара» с данными: ID инструмента, модель, наименование, фирма, страна производства, год производства, категория, стоимость за единицу, количество на складе, фото и панель навигации;
   3. Администратор заполняет все поля;
   4. Администратор нажимает кнопку «Добавить» на панели навигации;
   5. Система добавляет новый товар в «Каталог инструментов» в соответствующую ему группу и закрывает окно «Добавление товара».

**Детальные функциональные требования:**

***Авторизация***

|  |  |
| --- | --- |
| Вариант использования | Авторизация |
| Приоритет | **Важно** / Желательно / Необязательно |
| Триггер | Запуск программы |
| Предусловие | Система должна отобразить пользователю модальное окно «Авторизация» |
| Основной сценарий | Все поля должны быть заполнены.  Система должна проверить корректность заполненных полей.  Система должна войти в программу и закрыть окно «Авторизация». Доступ к разделам соответствует должности пользователя. |
| Альтернативный сценарий | В любой момент пользователь может отменить вход в программу путем нажатия кнопки «Закрыть». |
| Постусловие | Система открывает главное окно программы |

***Изменение данных для входа в программу***

|  |  |
| --- | --- |
| Вариант использования | Изменение данных для входа в программу |
| Приоритет | Важно / Желательно / **Необязательно** |
| Триггер | Нажатие кнопки «Редактировать» в окне «Сотрудники» |
| Предусловие | Система должна отобразить администратору модальное окно «Редактирование данных» |
| Основной сценарий | Изменение полей «Логин» и «Пароль».  Система должна проверить корректность заполненных полей.  Система должна сохранить изменения.  Администратор должен нажать на кнопку «Закрыть».  Система должна закрыть окно «Редактирование данных». |
| Альтернативный сценарий | В любой момент администратор может отменить редактирование путем нажатия кнопки «Отмена».  Система должна вернуть исходные данные и не обновлять «Логин» и «Пароль» для выбранного сотрудника. |
| Постусловие | В окне «Сотрудники» система должна обновить измененные данные. |

***Добавление покупателя***

|  |  |
| --- | --- |
| Вариант использования | Добавление пользователя |
| Приоритет | **Важно** / Желательно / Необязательно |
| Триггер | Нажатие кнопки «Добавить» в окне «Добавление продажи» |
| Предусловие | Система должна отобразить менеджеру модальное окно «Добавление пользователя» |
| Основной сценарий | Все поля должны быть заполнены.  Система должна проверить корректность заполненных полей.  Система должна добавить нового покупателя в список существующих в окне «Покупатели».  Система должна внести данные о новом покупателе в окно «Добавление продажи» в соответствующие поля.  Менеджер должен нажать на кнопку «Закрыть».  Система должна закрыть окно «Добавление покупателя». |
| Альтернативный сценарий | Менеджер может отменить редактирование путем нажатия кнопки «Отмена».  Система должна закрыть окно «Добавление покупателя» и не добавлять новые данные в окна «Покупатели» и поля в окне «Добавление продажи». |
| Постусловие | В окне «Добавление продажи» система должна внести новые данные в соответствующие поля.  Система должна обновить список покупателей в окне «Покупатели» и добавить нового. |

***Добавление сотрудника***

|  |  |
| --- | --- |
| Вариант использования | Добавление сотрудника |
| Приоритет | **Важно** / Желательно / Необязательно |
| Триггер | Нажатие кнопки «Добавить сотрудника» в окне «Сотрудники». |
| Предусловие | Система должна отобразить администратору модальное окно «Добавление сотрудника». |
| Основной сценарий | Все поля должны быть заполнены.  Система должна проверить корректность заполненных полей.  Система должна сохранить изменения.  Администратор должен нажать на кнопку «Закрыть».  Система должна закрыть окно «Добавление сотрудника». |
| Альтернативный сценарий | В любой момент администратор может отменить добавление данных путем нажатия кнопки «Отмена».  Система должна закрыть окно «Добавление сотрудника» и не обновлять список сотрудников в окне «Сотрудники». |
| Постусловие | В окне «Сотрудники» система должна обновить список сотрудников и добавить данные нового работника. |

***Добавление продажи***

|  |  |
| --- | --- |
| Вариант использования | Добавление продажи |
| Приоритет | **Важно** / Желательно / Необязательно |
| Триггер | Нажатие кнопки «Добавить» в окне «Продажи» |
| Предусловие | Система должна отобразить менеджеру модальное окно «Добавление продажи» |
| Основной сценарий | Все поля должны быть заполнены.  Система должна проверить корректность заполненных полей.  Система должна сохранить изменения.  Менеджер должен нажать на кнопку «Закрыть».  Система должна закрыть окно «Добавление продажи». |
| Альтернативный сценарий | В любой момент менеджер может отменить добавление продажи путем нажатия кнопки «Отмена».  Система должна закрыть окно «Добавление продажи» и не обновлять список продажи в окне «Продажи». |
| Постусловие | В окне «Продажи» система должна обновить данные.  Система должна добавить новую продажу в каталог существующих в окне «Продажи». |

***Добавление товара***

|  |  |
| --- | --- |
| Вариант использования | Добавление товара |
| Приоритет | **Важно** / Желательно / Необязательно |
| Триггер | Нажатие кнопки «Добавить» в окне «Каталог инструментов» |
| Предусловие | Система должна отобразить администратору модальное окно «Добавление инструмента» |
| Основной сценарий | Все поля должны быть заполнены.  Система должна проверить корректность заполненных полей.  Система должна сохранить изменения.  Администратор должен нажать на кнопку «Закрыть».  Система должна закрыть окно «Добавление инструмента». |
| Альтернативный сценарий | Администратор может отменить добавление инструмента путем нажатия кнопки «Отмена».  Система должна закрыть окно «Добавление инструмента» и не обновлять каталог товаров в окне «Каталог инструментов». |
| Постусловие | В окне «Каталог инструментов» система должна обновить данные.  Система должна добавить данные о новом товаре в каталог существующих в окне «Каталог инструментов». |

Контрольные вопросы

* + - 1. Кратко опишите основные этапы разработки программного продукта.

Основные этапы разработки:

- планирование проекта – постановка цели и задачи, выбор действий для их реализации, распределение имеющихся ресурсов;

- составление требований – формирование технического задания на разработку программного продукта, в котором описываются требования к будущему продукту;

- проектирование - моделирования теоретической основы будущего продукта;

- кодирование - перевод результатов проектирования в текст на языке программирования;

- тестирование - выполнение программы для выявления дефектов;

- сопровождение - внесение изменений в эксплуатируемое ПО.

2. Что представляют собой артефакты программного продукта?

Артефакт — это любой созданный искусственно элемент программной системы. К артефактам программного продукта относятся установочные файлы, исполняемые сценарии и конфигурационные файлы, файлы с исходными кодами и текстовыми сценариями, разные разворачиваемые данные, проектная и техническая документация.

3. Опишите основные стратегии разработки ПО.

Стратегии разработки: водопадная, инкрементная, эволюционная, RAD, V-Model.

Водопадная - линейная последовательность этапов конструирования.

Инкрементная ­- В начале процесса определяются все пользовательские и системные требования, оставшаяся часть конструирования выполняется в виде последовательности версий. Первая версия реализует часть запланированных возможностей, следующая версия реализует дополнительные возможности и т. д., пока не будет получена полная система.

Эволюционная - Система также строится в виде последовательности версий, но в начале процесса определены не все требования. Требования уточняются в результате разработки версий.

RAD (rapid application development model или быстрая разработка приложений) ­- разновидность инкрементной модели. В RAD-модели компоненты или функции разрабатываются несколькими высококвалифицированными командами параллельно, будто несколько мини-проектов. Временные рамки одного цикла жестко ограничены. Созданные модули затем интегрируются в один рабочий прототип.

V-model - направлена на тщательную проверку и тестирование продукта, находящегося уже на первоначальных стадиях проектирования. Стадия тестирования проводится одновременно с соответствующей стадией разработки.

4. Кратко охарактеризуйте системы принципов разработки ПО (PSP, TSP, CMM).

Personal Software Process - определяет требования к компетенциям разработчика. Согласно этой модели каждый программист должен уметь: учитывать время, затраченное на работу над проектом, учитывать найденные дефекты, классифицировать типы дефектов, оценивать размер задачи, осуществлять систематический подход к описанию результатов тестирования, планировать программные задачи и т.д.

Team Software Process - делает ставку на самоуправляемые команды численностью 3-20 разработчиков. Команды должны: установить собственные цели, составить свой процесс и планы, отслеживать работу, поддерживать мотивацию и максимальную производительность.

Capability Maturity Model - подразумевает модель зрелости процессов в компании. Различают следующие уровни CMM: начальный – характеризуется полным отсутствием документации и быстрыми изменениями продукта; воспроизводимый - частично воспроизводим и дает предсказуемый результат; определенный - характеризуется формализацией и планом улучшения процесса с течением времени; управляемый - процесс формализован и контролируется с помощью набора метрик, позволяющих измерить эффективность процесса; улучшенный - процесс постоянно улучшается путем применения технических и административных инноваций.

5. Состав персонала разработки программного продукта.

В Состав персонала разработки ПО входят: менеджер проекта (подбор кадров, формулирование и исполнение плана проекта, руководство командой и т.д.); программисты: ведущий разработчик (главный специалист по разработке ПО), ведущий программист (отвечает за отдельные функции продукта), рядовые программисты (работают над реализацией определённой функции ПО под руководством ведущего программиста); тестировщики (отвечают за составление и исполнение плана тестирования программы): ведущий тестировщик (составление плана тестирования и его исполнение, выбор инструментов для тестирования), инженер по автоматизации (создание автоматизированных тестовых заданий согласно плану), рядовой тестировщик (тестирование программы, проведение автоматизированных испытаний); группа разработчиков пользовательской документации (обеспечивает пользователя справочными материалами): ведущий разработчик т.д., рядовой разработчик т.д.; инженерные психологи; технологи по разработке ПО; группа технической поддержки и т.д.

6. Укажите основные метрики качества программного продукта.

К основным метрикам относятся: число строк кода, затраченное время, число тестов, плотность дефектов, число неустранённых дефектов, число компонентов системы.

7. В чём заключается процесс инспектирования проекта? Укажите принципы инспектирования.

Инспектирование состоит в проверке частей проекта (требований, результатов проектирования, программного кода и т. п.) на наличие дефектов. Принцип инспектирования может быть обобщен четырьмя правилами: вскрытие дефектов, участие коллег, распределение ролей, тщательная подготовка.

8. Приведите примеры функциональных и нефункциональных требований к программному продукту.

Функциональные требования - перечень сервисов, которые должна выполнять система. К ним относятся: бизнес-требования, пользовательские требования, функциональные требования.

Пример:

- Пользователь должен иметь возможность проводить поиск необходимых ему книг и документов или по всему множеству доступных каталожных баз данных или по определенному их подмножеству.

- Система должна предоставлять пользователю подходящее средство просмотра библиотечных документов.

- Каждый заказ должен быть снабжен уникальным идентификатором, который копируется в формуляр пользователя для постоянного хранения.

Нефункциональные требования - описывают характеристики системы и ее окружения, а не поведение системы. К ним относятся: бизнес-правила, внешние интерфейсы, атрибуты качества, ограничения.

9. Что представляет собой варианты использования программного продукта?

Прецеденты (Use cases) — спецификация последовательностей действий, которые может осуществлять система, подсистема или класс, взаимодействуя с внешними действующими лицами. Степень формализма выполняемого проекта и стадия, на которой он находится, прямо влияют на степень детальности и проработанности вариантов использования.

10. Приведите схему процесса анализа C-требований.

Порядок составления C-требований:

1) Идентификация пользователей системы;

2) Проведение интервью с представителями заказчика (определение желаний, документирование требований, эскиз графического интерфейса);

3) Оформление C-требований;

4) Проверка и анализ C-требований.

11. Приведите схему процесса анализа D-требований.

Порядок составления D-требований:

1) Выбор системы организации D-требований;

2) Создание диаграммы последовательности для каждого варианта использования;

3) Получение D-требования из C-требований и от заказчика;

4) Подготовка черновика плана тестирования;

5) Утверждение требований заказчиком;

6) Печать требований.