

ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ
Физико – технический факультет
Кафедра «Информационно – вычислительная техника»

Лабораторная работа № 1
по ТРПО

Студент 1(уск.) курса ИВТ
Группа ИВТ 8
Очная форма обучения
Григорович Константин Александрович
Преподаватель
Савенков Иван Николаевич

Донецк - 2020

ЛАБОРАТОРНАЯ РАБОТА №1

Курс «Технологии разработки программного обеспечения»

Тема: Составление спецификаций требований заказчика программного продукта.

Цель: Научиться формализовать требования заказчика программного продукта и оформлять техническое задание на создание программного продукта.

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.

Тема: Информационная система гостиничного комплекса.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	4
1.1 Цели и задачи.....	4
Области применения	4
Определения, термины, сокращения.....	4
1.4 Ссылки	4
1.5 Обзор.....	5
ОБЩЕЕ ОПИСАНИЕ.....	6
2.1 Перспективы продукта (позиционирование).....	6
2.1.1 Пользовательские интерфейсы	6
2.1.2 Требования к пользовательским интерфейсам	7
2.1.3 Требования к аппаратным интерфейсам	8
2.1.4 Требования к программным интерфейсам	8
2.1.5 Требования к коммуникационным интерфейсам	8
2.1.7 Операции.....	8
2.1.8 Требования к адаптации на месте.....	8
2.2 Требования к пользователям продукта	8
Ограничения	9
Предположения и зависимости	9
3. Детальные требования	10
3.1 Варианты использования	10
3.2.1 Зачисление клиента	10
3.2.2. Проверка корректности данных клиента	11
3.3 Нефункциональные требования	12
3.3.1. Требования к пользовательскому интерфейсу.....	12
3.3.1.1. Окно «Добавление клиента»	12
3.3.2. Требования к текстам сообщений об ошибках	12
3.3.2.1. Ошибки в данных клиента	13
3.3.3. Требования к данным	13
3.3.3.1. Требования к данным о пациентах	13

ВВЕДЕНИЕ

1.1 Цели и задачи

Цель проекта – создание информационной системы гостиничного комплекса

Задачи проекта:

- определение целевой архитектуры ИТ-комплекса гостиницы;
- определение оптимальной номенклатуры информационных систем (ИС), необходимых для поддержки основных, вспомогательных и обеспечивающих бизнес-процессов гостиницы;
- определение функциональных требований к информационным системам гостиницы;
- определение основных технических требований к информационным системам гостиницы;
- определение возможностей модернизации имеющихся информационных систем для интеграции их в создаваемый ИТ-комплекс гостиницы и удовлетворения функциональным и техническим требованиям;
- предварительное определение возможных рекомендуемых поставщиков и продуктов для информационных систем, соответствующих функциональным и техническим требованиям и доступным на рынке для запроса от них коммерческих предложений на поставку и внедрение информационных систем;
- определение приоритетов и последовательности работ при создании ИТ-комплекса гостиницы в соответствии с целевой архитектурой.

1.2 Области применения

Данная информационная система создана для обработки и хранения информации.

1.3 Определения, термины, сокращения

БД – База данных

ИС-Информационная система

IEEE Std. 830-1998 – стандарт спецификаций требований

SRS – Software Requirements Specification – структурированный набор требований к разрабатываемому программному продукту

1.4 Ссылки

Стандарт организации IEEE по составлению тех. заданий на разработку ПО: IEEE Std. 830-1998 IEEE Recommended Practice for Software Requirements Specification. IEEE Computer Society, 1998.

1.5 Обзор

Во втором разделе документа приведены общие требования к разрабатываемому продукту (концепция интерфейса и функционал). Целевой аудиторией раздела является – заказчик. В третьем разделе приводятся детальные требования к продукту, написанные разработчиком в результате подробного анализа общих требований.

ОБЩЕЕ ОПИСАНИЕ

Настоящий документ предназначен для определения общих принципов и подходов к развитию комплекса информационных и коммуникационных технологий (ИКТ) в ходе создания

Предполагается, что опыт развития ИКТ-комплекса должен быть в дальнейшем распространен на все гостиницы холдинга.

Настоящий документ содержит концептуальные положения и не предназначен для описания конкретных решений. Конкретные технические решения должны быть предложены в рамках работ по подготовке технико-экономических обоснований в области:

- создания ИС-комплекса гостиницы (комплекса информационных систем, поддерживающих основные, вспомогательные и обеспечивающие бизнес-процессы гостиницы);
- создания ИКТ-инфраструктуры гостиницы, обеспечивающей функционирование ИС-комплекса;
- создания комплекса дополнительных услуг гостям отеля, обеспечивающего предоставление платных и бесплатных телекоммуникационных и мультимедийных услуг в номерах и других помещениях;
- создания комплекса технических средств обеспечения физической безопасности в гостинице;
- создания комплекса обеспечения информационной безопасности и защиты персональных данных в гостинице.

2.1 Перспективы продукта (позиционирование)

2.1.1 Пользовательские интерфейсы

Программная система должна вести нижеуказанную информацию по следующим объектам:

Клиенты	Сотрудники	Ведомости
ФИО	ФИО	Дата, и время поступления
Дата рождения	Должность	Время нахождения
Ид. номер	Ид. номер	Дата, и время отъезда
Сканы документов	Ученая степень	Счет
Жалобы	Стаж	

С системой могут работать 2 группы пользователей:

- администраторы

-сотрудники

Обычным пользователям и администраторам должны предоставляться отдельные приложения. Администраторы могут также работать с модулем обработки журнала событий. Это приложение может быть запущено как автономно, так и из приложения администратора. Функциональная схема ПО может быть оформлена в упрощенном виде, как в примере на рис.1.

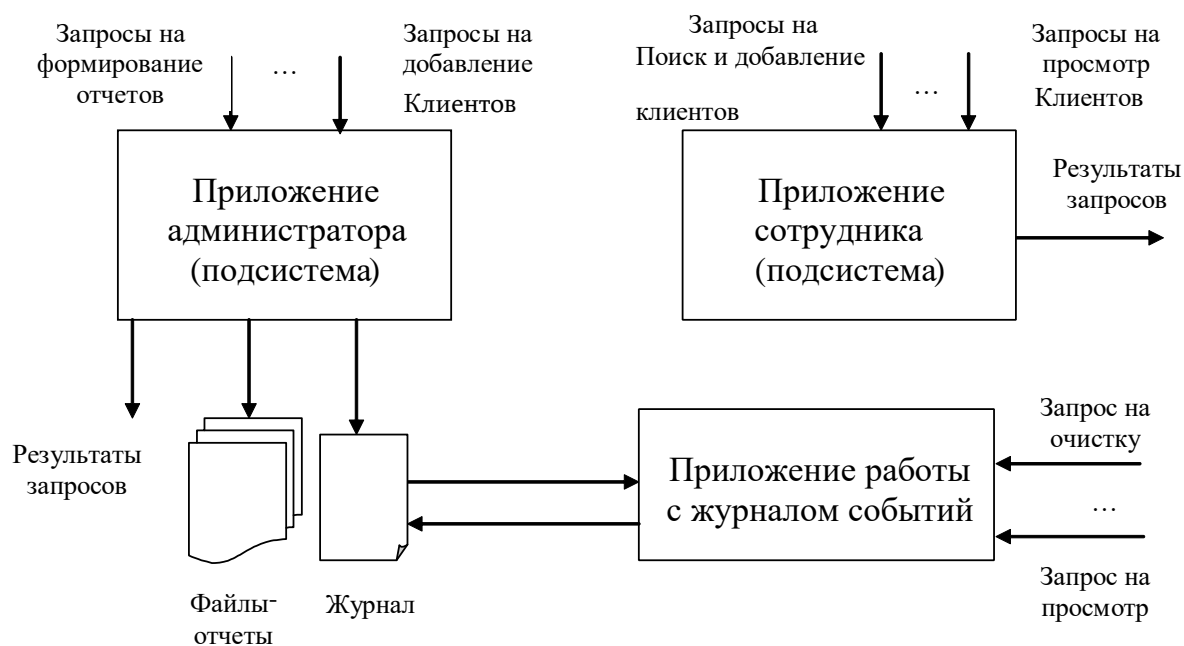


Рисунок 1 – Схема функциональной структуры системы «Больница»

2.1.2 Требования к пользовательским интерфейсам

Интерфейс должен предполагать стандартную системную цветовую палитру и разрабатываться под разрешение экрана 800x600. Оконные элементы управления и размер шрифта должны быть несколько больше стандартных размеров (оператор будет находиться относительно далеко от монитора). Окна должны обладать системным меню с кнопкой закрытия. Активное текущее окно должно не давать возможности работать с другими окнами программы.

ГЛАВНОЕ ОКНО, в котором должны быть вкладки «Документы», «Сотрудники», «Пропуски». В каждой вкладке отображаются списки и информация по выбранным документам, сотрудникам и пропускам, соответственно. На каждой вкладке должны быть кнопки «Изменить», «Добавить», «Удалить» для редактирования данных. В главном окне должно быть также меню с пунктом «Настройки».

...

2.1.3 Требования к аппаратным интерфейсам

Необходима поддержка сканера ШК, подключаемого к компьютеру через COM-порт.

2.1.4 Требования к программным интерфейсам

Необходимо обеспечить программное взаимодействие системы:
с системой управления БД PostgreSQL 8.4.
с операционными системами Windows 7,8.10.

2.1.5 Требования к коммуникационным интерфейсам

Необходима поддержка сетевого протокола TCP/IP для обмена сообщений между модулями системы.

2.1.6 Требования к ПАМЯТИ

Программа должна занимать не более 500 Мб оперативной памяти.
Модули программы должны занимать не более 2000 Мб памяти на жестком диске.

2.1.7 Операции

Необходима поддержка следующих операций:
резервное копирование (бекап) БД
конфигурирование БД и настроек подключения
сканирование ШК
отправка уведомлений на адрес электронной почты
варьирование электронных документов (создание ревизий)

2.1.8 Требования к адаптации на месте

Необходимы программы-установщики для осуществления развертывания модулей администратора, архивариуса, актуализаторов и директора на соответствующих целевых компьютерах. Необходимо предоставить всем категориям пользователей справочную информацию.

2.2 Требования к пользователям продукта

Администратор:

Понимание принципов функционирования систем управления БД

Архивариус:

Доступ к архиву документов

Навыки работы со сканером ШК

Директор:

Владение компьютером на уровне пользователя ОС Windows Клиент:

Владение компьютером на уровне пользователя ОС Windows

Все пользователи, кроме класса «Клиент», должны быть сотрудниками предприятия.

Ограничения

Должна использоваться кодировка UTF-8.

Продукт будет поддерживать только русский язык пользовательского интерфейса.

Продукт не предусматривает автоматического перехода на платформы, не перечисленные в данном документе.

Продукт не будет корректно вести историю версий документов, если заказчик будет вручную удалять или изменять имена файлов документов, хранящихся в БД.

Предположения и зависимости

Скорость передачи изображений будет определяться только состоянием сети.

Скорость выполнения запросов к БД будет определяться производительностью и степенью загрузки системы управления БД.

3. Детальные требования

3.1 Варианты использования

Вариант использования «Зачислить клиента»:

- 1) Администратор в окне с информацией о клиентах нажимает кнопку «Добавить»
- 2) Система отображает окно «Добавление клиента» с полями для ввода ФИО, даты рождения, ид.номера, Жалоб, Скана документов, и с кнопками «ОК» и «Отмена».
- 3) Администратор в появившемся окне заполняет все поля.
- 4) Администратор нажимает кнопку «ОК».
- 5) Система закрывает окно «Добавление клиента».
- 6) Система добавляет в список нового клиента и устанавливает выделение на нем.

Вариант использования «Сформировать отчет по клиентам»:

- 7) Администратор в окне отчетов, в выпадающем списке с типами отчета, выбирает «Отчет по клиентам».
- 8) Администратор нажимает кнопку «Сформировать».
- 9) Система отображает окно с выбором пути для сохранения файла-отчета и 2 кнопками «ОК» и «Отмена».
- 10) Администратор указывает путь для сохранения файла-отчета и нажимает кнопку «ОК». Если Администратор нажимает кнопку «Отмена», сценарий останавливается.
- 11) Система создает текстовый файл с выбранным именем, содержащий отчет.
- 12) Система отображает окно с вопросом «Просмотреть отчет?» и 2 кнопками «Да» и «Нет».
- 13) Если Администратор нажимает «Да», то Система отображает содержимое файла.

Примеры детальных функциональных требований.

Ниже приводится пример организации детальных функциональных требований по вариантам использования. Требования должны быть верифицируемыми, т.е. однозначно проверяемыми. Все требования должны быть пронумерованы (промаркированы).

3.2.1 Зачисление клиента

Вариант использования	Зачислить пациента
-----------------------	--------------------

Приоритет	<u>Важно</u> / Желательно / Необязательно
Триггер	Нажатие кнопки «Добавить пациента» в окне «Пациенты»
Предусловие	1. Система должна отобразить администратору модальное окно «Добавление пациента».
Основной сценарий	<p>Все поля должны быть доступны для редактирования, а кнопки «ОК» и «Отмена» должны быть активными.</p> <p>Система должна проверить корректность заполненных данных (см. 3.2.36).</p> <p>Система должна добавлять запись типа «Зачисление Пациента» в лог. Дата и время должны быть равны системной дате и времени.</p> <p>Система должна закрыть окно «Добавление пациента».</p>
Альтернативный сценарий	6. В любой момент администратор может отменить зачисление пациента путем нажатия на кнопку «Отмена». Система должна закрыть окно «Добавление пациента» и не обновлять список пациентов в окне «Пациенты».
Постусловие	7. В окне «Пациенты» система должна обновить список Пациентов и установить выделение на добавленном Пациенте.
Сценарий исключительных ситуаций	8. Система должна генерировать исключение при проблеме с записью в файл лога.
Ссылки	Пункты SRS (ТЗ): 2.3.17 3.1.1.5, 3.2.36

3.2.2. Проверка корректности данных клиента

Вариант использования	Проверить корректность данных пациента
Приоритет	<u>Важно</u> / Желательно / Необязательно
Триггер	Вариант использования «Добавить пациента»
Предусловие	Активно модальное окно «Добавление пациента»
Основной сценарий	Все поля должны быть заполнены. Значения всех полей должны соответствовать требованиям <u>3.1.3.1</u> . Система должна проверить дублирование ид.номера.
Альтернативный сценарий	4. Если любая из проверок окончилась неудачей, все поля в окне «Добавление пациента» очищаются, а остальные проверки не производятся. Администратору выводится окно с сообщением об ошибке (п. <u>3.1.2.1</u>).
Постусловие	-
Сценарий исключительных ситуаций	-
Ссылки	Пункты SRS (ТЗ): 2.3.17 3.1.1.5, 3.1.2.1 Пункты STD (Тесты): 1.36

Примечание. Для данного сценария разрабатываются также тесты.

3.3 Нефункциональные требования

3.3.1. Требования к пользовательскому интерфейсу

Простота в эксплуатации

доступность

3.3.1.1. Окно «Добавление клиента»

<картинка> или <текстовое подробное описание>

3.3.2. Требования к текстам сообщений об ошибках

Понятность, из-за чего ошибка

3.3.2.1. Ошибки в данных клиента

Ошибка	Текст сообщения
xxxxxx0n (не заполнено поле)	Необходимо заполнить все поля!
xxxxxx0n (значение некорректно)	Неверное значение поля <поле> поля
xxxxxx0n (дублирование ид.номера)	Такой пациент уже присутствует в базе! Идентификационный номер: <idN>, ФИО: <fio>

3.3.3. Требования к данным

Краткость

Понятность

3.3.3.1. Требования к данным о пациентах

Данные	Требования
Фамилия	Произвольная строка
Имя	Произвольная строка
Отчество	Произвольная строка
Дата рождения	Строка в формате «dd.mm.yyyy»
Ид. номер	Строка из 10 символов-цифр
Жалобы	Произвольная строка

Контрольные вопросы:

1. Кратко опишите основные этапы разработки программного продукта.

Анализ требований ...

Проектирование ...

Кодирование ...

Тестирование и отладка ...

Внедрение ...

Заключение

2. Что представляют собой артефакты программного продукта?

Результаты процесса разработки программного продукта, которые напрямую не участвуют в работе созданного приложения, но являются результатами разработки, например — файлы исходного кода и файлы данных, из которых создаются артефакты размещения.

3. Опишите основные стратегии разработки ПО.

Каскадная:

Однократный проход всех этапов разработки

Инкрементная:

Многokrатный проход этапов разработки с запланированным улучшением результата

Эволюционная:

Многokrатный проход этапов разработки

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

Одна из последних разработок Института программной инженерии Personal Software Process / Team Software Process . 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;
- распределять их по времени и составлять график работы.
- выполнять индивидуальную проверку проекта и архитектуры;
- осуществлять индивидуальную проверку кода;
- выполнять регрессионное тестирование.

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.

Последовательное применение модели PSP/TSP позволяет сделать нормой в организации пятый уровень CMM.

CMM (Capability Maturity Model) переводится как «модель зрелости возможностей», а SMM (Social Media Marketing) — маркетинг в социальных медиа. Подразумевает модель зрелости процессов в компании.

5. Состав персонала разработки программного продукта.

6. Укажите основные метрики качества программного продукта.

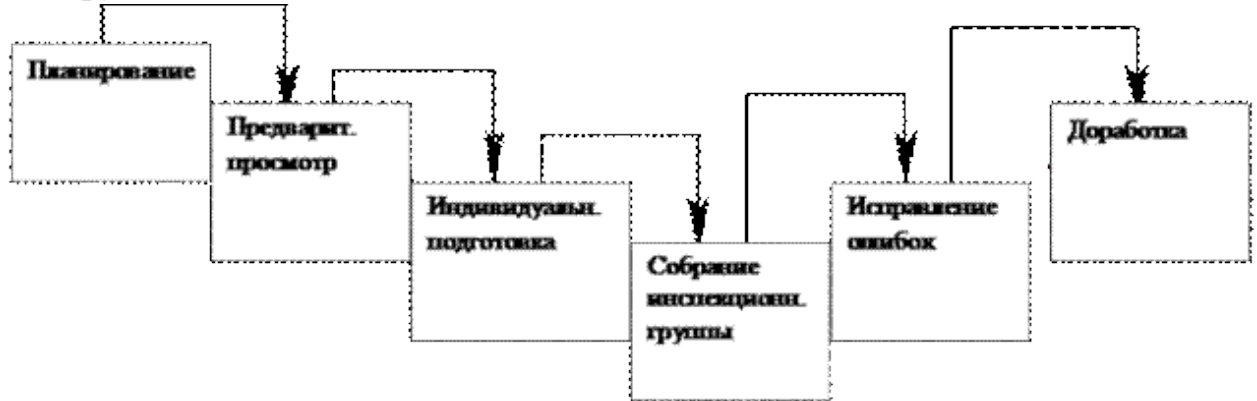
Метрика качества программ — система измерений качества программ. Измерения характеристик можно выполнить объективно и достоверно. Однако не следует исключать того, что оценка качества ПО в целом может быть связана с субъективной интерпретацией получаемых оценок.

Все метрики ПО разделяются на два класса:

- а) метрики, характеризующие наиболее специфические свойства программ, т.е. метрики оценки качества самого ПО;
- б) метрики оценки технических характеристик и факторов разработки программ, т.е. метрики оценки условий разработки программ

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

Инспектирование программ – это просмотр и проверка программ с целью обнаружения в них ошибок.



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

Функциональное требование описывает, что должна делать программная система, в то время как нефункциональные требования накладывают ограничения на то, как система будет это делать.

Примером функционального требования может служить:

Система должна посылать email всякий раз, когда выполняется определенное условие (например, заказ размещен, клиент регистрируется и т. д.).

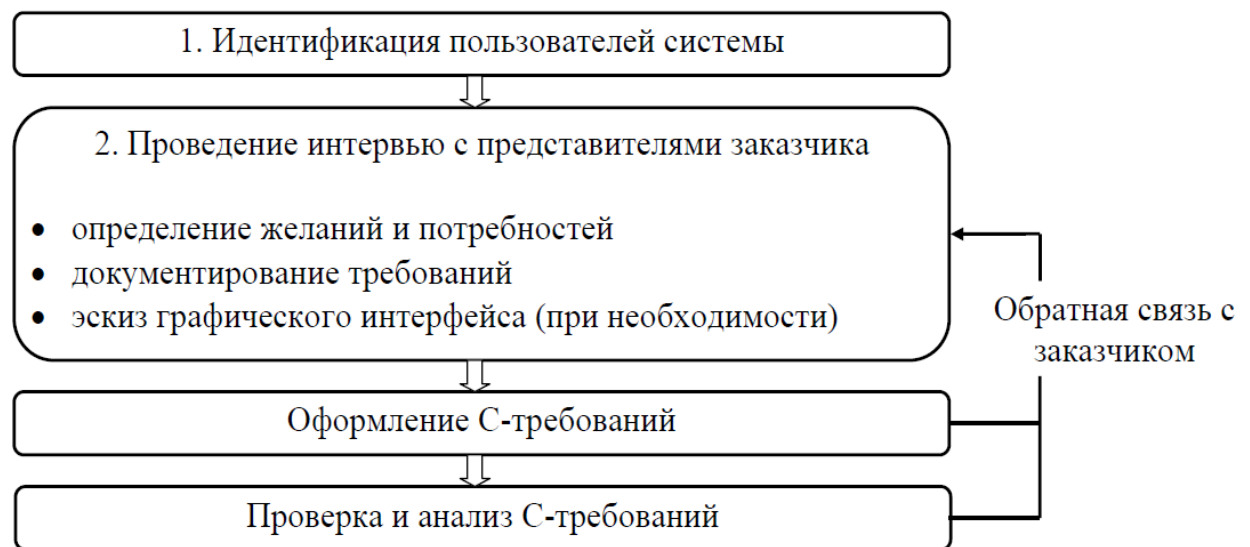
Соответствующим нефункциональным требованием к системе может быть::

Электронные письма должны быть отправлены с задержкой не более 12 часов с момента такой активности.

9. Что представляют собой варианты использования программного продукта?

Варианты использования это - описание последовательности действий, которые может осуществлять система в ответ на внешние воздействия пользователей или других программных систем

10. Приведите схему процесса анализа С-требований.



11. Приведите схему процесса анализа D-требований.

