**1.Постановка задачи.**

**Разработать ПО ИС аптеки:**

1) с применением структурного подхода, создав: начальную контекстную диаграмму; концептуальную модель данных с атрибутами; диаграммы потоков данных нулевого и последующих уровней для процессов ИС; диаграммы системных процессов нулевого и последующих уровней; диаграмму последовательности экранных форм.

2) с применением объектно-ориентированного подхода в среде Rational Rose реализовать: диаграмму вариантов использования; диаграмму классов; диаграмму последовательности; кооперативную диаграмму; диаграмму пакетов; сетевую конфигурацию системы; диаграмму состояния.

**2. Структурный подход к разработке ПО ИС Аптеки.**

**2.1. Жизненный цикл ПО ИС.**

Одним из базовых понятий методологии проектирования ИС является понятие жизненного цикла ее программного обеспечения (ЖЦ ПО). ЖЦ ПО - это непрерывный процесс, который начинается с момента принятия решения о необходимости его создания и заканчивается в момент его полного изъятия из эксплуатации.

Основным нормативным документом, регламентирующим ЖЦ ПО, является международный стандарт ISO/IEC 12207 [5] (ISO - International Organization of Standardization - Международная организация по стандартизации, IEC - International Electrotechnical Commission - Международная комиссия по электротехнике). Он определяет структуру ЖЦ, содержащую процессы, действия и задачи, которые должны быть выполнены во время создания ПО.

Структура ЖЦ ПО по стандарту ISO/IEC 12207 базируется на трех группах процессов:

· основные процессы ЖЦ ПО (приобретение, поставка, разработка, эксплуатация, сопровождение);

· вспомогательные процессы, обеспечивающие выполнение основных процессов (документирование, управление конфигурацией, обеспечение качества, верификация, аттестация, оценка, аудит, решение проблем);

· организационные процессы (управление проектами, создание инфраструктуры проекта, определение, оценка и улучшение самого ЖЦ, обучение).

Разработка включает в себя все работы по созданию ПО и его компонент в соответствии с заданными требованиями, включая оформление проектной и эксплуатационной документации, подготовку материалов, необходимых для проверки работоспособности и соответствующего качества программных продуктов, материалов, необходимых для организации обучения персонала и т.д. Разработка ПО включает в себя, как правило, анализ, проектирование и реализацию (программирование).

Эксплуатация включает в себя работы по внедрению компонентов ПО в эксплуатацию, в том числе конфигурирование базы данных и рабочих мест пользователей, обеспечение эксплуатационной документацией, проведение обучения персонала и т.д., и непосредственно эксплуатацию, в том числе локализацию проблем и устранение причин их возникновения, модификацию ПО в рамках установленного регламента, подготовку предложений по совершенствованию, развитию и модернизации системы.

Управление проектом связано с вопросами планирования и организации работ, создания коллективов разработчиков и контроля за сроками и качеством выполняемых работ. Техническое и организационное обеспечение проекта включает выбор методов и инструментальных средств для реализации проекта, определение методов описания промежуточных состояний разработки, разработку методов и средств испытаний ПО, обучение персонала и т.п. Обеспечение качества проекта связано с проблемами верификации, проверки и тестирования ПО. Верификация - это процесс определения того, отвечает ли текущее состояние разработки, достигнутое на данном этапе, требованиям этого этапа. Проверка позволяет оценить соответствие параметров разработки с исходными требованиями. Проверка частично совпадает с тестированием, которое связано с идентификацией различий между действительными и ожидаемыми результатами и оценкой соответствия характеристик ПО исходным требованиям. В процессе реализации проекта важное место занимают вопросы идентификации, описания и контроля конфигурации отдельных компонентов и всей системы в целом.

Управление конфигурацией является одним из вспомогательных процессов, поддерживающих основные процессы жизненного цикла ПО, прежде всего процессы разработки и сопровождения ПО. При создании проектов сложных ИС, состоящих из многих компонентов, каждый из которых может иметь разновидности или версии, возникает проблема учета их связей и функций, создания унифицированной структуры и обеспечения развития всей системы. Управление конфигурацией позволяет организовать, систематически учитывать и контролировать внесение изменений в ПО на всех стадиях ЖЦ. Общие принципы и рекомендации конфигурационного учета, планирования и управления конфигурациями ПО отражены в проекте стандарта ISO 12207-2 [5].

Каждый процесс характеризуется определенными задачами и методами их решения, исходными данными, полученными на предыдущем этапе, и результатами. Результатами анализа, в частности, являются функциональные модели, информационные модели и соответствующие им диаграммы. ЖЦ ПО носит итерационный характер: результаты очередного этапа часто вызывают изменения в проектных решениях, выработанных на более ранних этапах.

**Модели жизненного цикла ПО**

Стандарт ISO/IEC 12207 не предлагает конкретную модель ЖЦ и методы разработки ПО (под моделью ЖЦ понимается структура, определяющая последовательность выполнения и взаимосвязи процессов, действий и задач, выполняемых на протяжении ЖЦ. Модель ЖЦ зависит от специфики ИС и специфики условий, в которых последняя создается и функционирует). Его регламенты являются общими для любых моделей ЖЦ, методологий и технологий разработки. Стандарт ISO/IEC 12207 описывает структуру процессов ЖЦ ПО, но не конкретизирует в деталях, как реализовать или выполнить действия и задачи, включенные в эти процессы.

К настоящему времени наибольшее распространение получили следующие две основные модели ЖЦ:

· каскадная модель (70-85 г.г.);

· спиральная модель (86-90 г.г.).

В изначально существовавших однородных ИС каждое приложение представляло собой единое целое. Для разработки такого типа приложений применялся каскадный способ. Его основной характеристикой является разбиение всей разработки на этапы, причем переход с одного этапа на следующий происходит только после того, как будет полностью завершена работа на текущем. Каждый этап завершается выпуском полного комплекта документации, достаточной для того, чтобы разработка могла быть продолжена другой командой разработчиков.

Положительные стороны применения каскадного подхода заключаются в следующем [2]:

· на каждом этапе формируется законченный набор проектной документации, отвечающий критериям полноты и согласованности;

· выполняемые в логичной последовательности этапы работ позволяют планировать сроки завершения всех работ и соответствующие затраты.

**2.1.3. Стандарт ISO 12207.**

Стандарт ISO 12207 — Процессы жизненного цикла программного обеспечения — наиболее полно на уровне международных стандартов отражает жизненный цикл, технологию разработки и обеспечения качества сложных программных средств. Жизненный цикл ПО представлен набором этапов, частных работ и операций в последовательности их выполнения и взаимосвязи, регламентирующих ведения разработки на всех стадиях от подготовки технического задания до завершения испытаний ряда версий и окончания эксплуатации ПО. В ЖЦ включаются описания исходной информации, способов выполнения операций и работ, устанавливаются требования к результатам и правилам их контроля, а также к содержанию технологических и эксплуатационных документов. Определяется организационная структура коллективов, распределение и планирование работ, а также контроль за реализацией ЖЦ ПО.

Стандарт определяет архитектуру, процессы, разделы и подразделы ЖЦ ПО, а также перечень базовых работ и детализирует содержание каждой из них. Архитектура ЖЦ ПО в стандарте базируется на трех крупных компонентах (Рис. 4):

* основные процессы жизненного цикла ПО и определяющие работы (раздел 5);
* вспомогательные процессы и работы, поддерживающие жизненный цикл ПО (раздел 6);
* организационные процессы и управление жизненным циклом ПО (раздел 7).

Эти разделы стандарта состоят из ряда подразделов, в которых подробно раскрывается содержание каждой работы и комментируются особенности их выполнения. Рекомендации к каждому подразделу состоят в среднем из 3-6 пунктов — работ (процедур). Общее число работ и комментариев к ним в стандарте свыше 220.

В разделе 5 изложены основы ЖЦ и рекомендации по подготовке, разработке, эксплуатации и сопровождению программных средств (см. Рис. 4). Процессы приобретения и/или подготовки к созданию ПО должны начинаться с инициализации проекта, анализа концепции, анализа рынка продуктов, выработки требований и состава поддерживающих документов, создания предварительного плана проекта. Основные работы по созданию сложного комплекса программ рекомендуется начинать с определения состава сопровождающих документов, выбора средств конфигурационного управления и обеспечения качества, а также выбора методов и средств технологического обеспечения разработки всей информационной системы. Кодирование и тестирование каждого компонента ПО должно быть оформлено совокупностью документов, удостоверяющих соответствие компонента первичной спецификации, содержащих тесты и результаты тестирования.

Рекомендуется разрабатывать план работ, включающий комплексирование компонентов, тестирование по всем разделам требований и показателям качества, а также документирование плана, результатов интеграции, использованных тестов, критериев оценки и полученных результатов. Далее ПО следует подвергать квалификационному (аттестационному) тестированию по всем разделам требований контракта, при широком варьировании тестов, изменениях значений критериев, а также тестировать полноту и адекватность технологической и пользовательской документации реальному программному продукту. Проверенный таким образом комплекс программ интегрируется в вычислительные средства информационной системы, средства визуализации и телекоммуникации.

Эти работы взаимодействуют с работами, обеспечивающими сопровождение ПО. Специалисты анализируют сообщения об ошибках и предложения на модификацию ПО, селектируют их на соответствие требованиям контракта и оценивают целесообразность проведения изменений. Подготовленные изменения тестируются и проверяются по критериям, определенным в документации.

Вспомогательные технологические работы, поддерживающие жизненный цикл ПО, и рекомендации по их выполнению изложены в разделе 6. Процессы документирования ПО должны охватывать планирование и обеспечение документирования, рекомендации по стандартизации, проектированию и разработке, а также по производству, конфигурационному управлению и сопровождению комплекта документации на ПО. Для обеспечения гарантий качества следует использовать планирование, методологию, процедуры и стандарты поддержки качества ПО в соответствии с контрактом с учетом доступных ресурсов. Верификация ПО должна включать ее организацию, планирование и техническое обеспечение. Удостоверение правильности (аттестация) должна гарантировать полное соответствие программного продукта спецификациям, требованиям и документации на ПО и возможность его надежного функционирования и безопасного применения пользователем.

Организации жизненного цикла ПО посвящен раздел 7. Она включает основные работы по управлению проектом, производством и средствами для обеспечения процессов по разработке, эксплуатации и сопровождению. Процессы формирования инфраструктуры должны состоять из выбора и установления аппаратных и программных средств, технологии, стандартов и обслуживания, используемых для разработки, сопровождения и обеспечения эксплуатации ПС. Процессы совершенствования жизненного цикла ПС состоят в установлении, оценивании, измерении, контроле и корректировке процессов жизненного цикла конкретных ПО. Процессы обучения определяются требованиями к проекту, должны учитывать необходимые ресурсы, управление и технические средства. Изложены рекомендации по преобразованию и адаптации базовой структуры этого международного стандарта для конкретного проекта (приложение А) и руководство по их выполнению в ЖЦ ПО (приложение В).

**2.2. Диаграммы, реализованные в структурном подходе.**

1. **Начальная контекстная диаграмма ПО ИС Аптеки:**

1|\_\_\_\_\_\_\_\_\_\_\_\_\_

Покупатель

2|\_\_\_\_\_\_\_\_\_\_\_\_

Менеджер

Чек Документы

Полученное лекарство Заказ (опт)

База лекарств

Заявка на лекарство ТТН

0

\_\_\_\_\_\_\_ИС Аптеки\_\_\_\_\_\_\_\_\_\_

Заказ от Зав. Аптеки

Заявка на лекарство Документы

База лекарств

Лекарство Отправка заказа на лекарства

Чек ТТН

Заказ (опт)

Наличие данного лекарства

4|\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Зав. Аптекой

3|\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кассир

1. **Концептуальная модель данных с атрибутами.**

(0,N) (1,1)

Менеджер

Заведующий аптекой

(1,1) (0,N

(0,N)

\_\_\_\_\_\_\_\_\_Заказ\_\_\_\_\_\_\_\_\_\_\_

Идентификационный №

Наименование лекарства

Фирма-производитель

Страна

Стоимость

Количество

(0,N)

(1,1)

\_\_\_\_\_Лекарство\_\_\_\_\_\_\_\_\_\_

Идентификационный №

Наименование лекарства

Фирма-производитель

Страна

Стоимость

Количество

(1,1)

(N,N)

(1,1)

(1,1)

(1,1)

(1,1)

(0,N) (1,1)

Покупатель

(1,1)

Кассир

1. **Диаграмма потоков 0-го уровня.**

2|\_\_\_\_\_\_\_\_\_\_\_\_\_

Менеджер

1|\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Покупатель

Оплата лекарства

Документы для получения заказа

Товарно-транспортная накладная

Отправка заказа на лекарства

Полученное лекарство

База лекарств

Заявка на лекарство

Чек

Заказ(опт)

1 База лекарств

\_\_\_\_\_\_\_\_\_\_\_\_3\_\_\_\_\_\_\_\_\_\_\_\_\_

Администрирование

\_\_оплаты и выдачи лекарств\_

\_\_\_\_\_\_\_\_\_\_\_\_1\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Администрирование

\_\_\_\_\_\_\_\_\_\_заказов\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_2\_\_\_\_\_\_\_\_\_\_\_\_\_

Администрирование

\_\_\_\_\_\_приема заявок\_\_\_\_\_\_\_

Товарно-транспортная накладная

Отправка заказа на лекарства

Документы для получения заказа

Наличие лекарства

База лекарств

Заявка на лекарство

4|\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Зав. Аптекой

Заказ(опт)

Оплата лекарства

Лекарство

3|\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кассир

Чек

**4. Диаграмма потоков данных первого уровня для процесса 2.**

1|\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Покупатель

Заявка на лекарство

\_\_\_\_\_\_\_\_\_\_\_\_2.1\_\_\_\_\_\_\_\_\_\_\_\_

Обслужить

\_\_\_\_\_\_\_\_покупателя\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_2.2\_\_\_\_\_\_\_\_\_\_\_\_

Подготовить

\_\_\_\_заявку от покупателя\_\_\_\_

Покупатели

База лекарств

Заявка на лекарство

Наличие лекарства

Лекарства

3|\_\_\_\_\_\_\_\_\_\_\_\_\_

Кассир

**5. Диаграмма потоков данных второго уровня для процесса 2.1.**

Заявка на лекарство

\_\_\_\_\_\_\_\_\_\_\_\_2.1.2\_\_\_\_\_\_\_\_\_\_

Зарегистрировать

\_\_\_\_\_заявку покупателя\_\_\_\_\_

1|\_\_\_\_\_\_\_\_\_\_\_\_\_

Покупатель

\_\_\_\_\_\_\_\_\_\_\_\_2.1.1\_\_\_\_\_\_\_\_\_\_

Подтвердить

\_\_\_\_\_\_\_\_регистрацию\_\_\_\_\_\_\_\_

Номер покупателя

Подтверждение регистрации

Покупатели

**6. Диаграмма потоков данных первого уровня для процесса 2**

Чек

Лекарство

Лекарство

3|\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кассир

\_\_\_\_\_\_\_\_\_\_\_\_2.2\_\_\_\_\_\_\_\_\_\_\_\_

Выдать

\_\_\_\_\_\_\_\_\_лекарство\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_2.1\_\_\_\_\_\_\_\_\_\_\_\_

Оплатить

\_\_\_\_\_\_\_\_\_лекарство\_\_\_\_\_\_\_\_\_

1|\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Покупатель

Оплата

Чек

Лекарства

**7. Диаграмма системных процессов нулевого уровня**

База лекарств

Лекарство

Наличие данного лекарства

Стоимость лекарства

Оплата лекарства

2|\_\_\_\_\_\_\_\_\_\_\_\_\_

Менеджер

Заявка на лекарство

Полученное лекарство

Чек

База лекарств

Чек

5|\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Принтер

0

\_\_\_\_\_\_\_ИС Аптеки\_\_\_\_\_\_\_\_\_\_

4|\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Зав. Аптекой

3|\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кассир

1|\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Покупатель

Документы

Заказ (опт)

Заказ на лекарства

ТТН

Документы

Заказ на лекарства

ТТН

Заказ(опт)к



Чекк



**8. Диаграмма системных процессов первого уровня**

2|\_\_\_\_\_\_\_\_\_\_\_\_\_

Менеджер

5|\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Принтер

4|\_\_\_\_\_\_\_\_\_\_\_\_\_

Зав. Аптекой

1|\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Покупатель

7|\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Сеть Интернет

4

\_ПК Заведующего аптекой\_\_\_\_

3|\_\_\_\_\_\_\_\_\_\_\_\_\_

Кассир

3

\_\_\_\_ПК Менеджера\_\_\_\_\_\_\_\_\_

2

\_\_\_\_ПК Отдела продаж\_\_\_\_\_\_\_

1

\_\_\_\_Сервер БД ИС Аптеки\_\_\_\_

Чекк



6|\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ЛВС

1 БД ИС Аптеки

**9. Диаграмма последовательности экранных форм.**

**Лекарство**

**Зав. Аптеки**

**Кассир**

**Менеджер**

**Покупатель**

**Персонал**

**Заявка**

**Чек**

**Покупка**

**Система**

**аптеки**

**3. Объектно-ориентированный подход к разработке ПО ИС Аптеки**

1. Use - Case Diagram.



2. Sequence Diagram.



3.Traceabilities.



4. Сетевая конфигурация системы.



5. Collaboration Diagram.



**Заключение.**

В данной курсовой работе разработана информационная система аптеки в Case-средстве Rational Rose корпорации Rational.

В ходе работы была создана начальная контекстная диаграмма, которая является основной при построении диаграмм DFD, разбивающаяся на диаграммы потоков нулевого и последующих уровней для процессов ИС. Также была сконструирована концептуальная модель с атрибутами, т.е. диаграмма ERD, являющаяся прототипом базы данных, которая включает в себя базу данных заказов и лекарств в аптечной сети.

Для представления диаграмм на физическом уровне были созданы диаграммы системных процессов, отображающие взаимосвязь компьютеров, людей посредством ЛВС и сетей Интернет.

Структурный подход дает основу для создания диаграмм объектно-ориентированного подхода в среде Rational Rose.

Объектно-ориентированный подход включает в себя в первую очередь диаграмму вариантов использования, которая представляет из себя действующих лиц, которые участвуют в создании ИС Аптеки (менеджер оптовой фирмы, заведующий аптекой, кассир, покупатель) и связанные с их деятельностью варианты использования.

Из диаграммы вариантов использования вытекают диаграммы последовательностей, позволяющие разбивать и уточнять каждый вариант использования.

Следующим этапом объектно-ориентированного подхода является создание классов с соответствующими атрибутами (с определением стереотипов классов) и взаимодействие между классами.

Диаграмма сетевой конфигурации системы показывает, что менеджеру, заведующему аптекой, кассиру, покупателю необходимы компьютеры, выделенный сервер для хранения заявок от покупателя, документов для получения заказов, базу данных медицинских лекарств. Компьютеры менеджеров, а также покупателей, с сервером будут соединятся с помощью глобальной сети Интернет.

Соответственно, компьютеры заведующей и кассира могут быть соединены с помощью ЛВС, и тоже могут работать с сервером через сеть Интернет.

В дальнейшем для этой информационной системы должны быть созданы программистами пользовательские приложения отдельно для клиентов, кассира, менеджеров, заведующей аптеки.

**Список использованной литературы.**

1. Мироненкова Ж.В. Развитие информационных сетей в фармацевтической отрасли / Мироненкова Ж.В., Лозовая Г.Ф. // В сб.: Материалы Х Российского национального конгресса «Человек и лекарство». – М., 2003. – С. 12.

2. http://ref.ruscore.ru/rqref/part6/item7101.html

3. http://www.klubok.net/downloads2.html

4. http://inform-referats.narod.ru/informatic-all.htm

5. http://rational.com