

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ, НГУ)

Кафедра систем информатики

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА

Усова Яна Александровна

**Разработка структуры корпоративной сети медицинских
учреждений**

Направление подготовки 230100.62 ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ
ТЕХНИКА

Руководител ь

Исаева Н.А.
(фамилия , И., О.)
к.э.н., доцент
(уч.степень, уч.звание)

.....
(подпись, дата)

Автор

Усова Я.А.
(фамилия , И., О.)
ФИТ, 9202
(факультет, группа)

.....
(подпись, дата)

Новосибирск, 2013 г.

Содержание

ВВЕДЕНИЕ	3
1. ОПИСАНИЕ ПРОБЛЕМЫ И ПРЕДМЕТНОЙ ОБЛАСТИ	5
1.1 АКТУАЛЬНОСТЬ ПРОБЛЕМЫ.....	5
1.2 СОВРЕМЕННОЕ ПОЛОЖЕНИЕ	6
1.3 «ЭЛЕКТРОННАЯ РЕГИСТРАТУРА» НОВОСИБИРСКОЙ ОБЛАСТИ	7
2. РАЗРАБОТКА СТРУКТУРЫ СЕТИ	8
2.1 КОРПОРАТИВНАЯ СЕТЬ КАК ОБЪЕКТ ИССЛЕДОВАНИЯ	8
2.2 РОЛЬ КОРПОРАТИВНЫХ СЕТЕЙ В СОЗДАНИИ И РАЗВИТИИ ИТ- ИНФРАСТРУКТУРЫ.....	9
2.3 ОСОБЕННОСТИ ПРОЕКТИРОВАНИЯ КОРПОРАТИВНЫХ СЕТЕЙ.....	9
2.4 ПАРАМЕТРЫ КАЧЕСТВА КОРПОРАТИВНОЙ СЕТИ.....	11
2.5 ТРЕХУРОВНЕВАЯ ИЕРАРХИЧЕСКАЯ МОДЕЛЬ	12
2.6 СТРУКТУРА СЕТИ	14
3. ПРИЛОЖЕНИЕ ДЛЯ РАБОТЫ С РЕЦЕПТАМИ	18
3.1 СРЕДСТВА РАЗРАБОТКИ	19
3.2 РЕАЛИЗАЦИЯ ОСНОВНОЙ ФУНКЦИОНАЛЬНОСТИ	19
ЗАКЛЮЧЕНИЕ.....	24
СПИСОК ЛИТЕРАТУРЫ	25
ПРИЛОЖЕНИЕ А.....	27
ПРИЛОЖЕНИЕ Б.....	29
ПРИЛОЖЕНИЕ В.....	31
ПРИЛОЖЕНИЕ Г.....	33

Введение

Основой инфраструктуры современных предприятий являются корпоративные сети передачи данных, предоставляющие транспорт для передачи информации между разными приложениями информационных систем.

В последнее время мультисервисные корпоративные сети приходят на смену специализированным сетям. Для обеспечения потребностей требования к мультисервисной корпоративной сети, непрерывно возрастают, как к среде передачи информации для выполнения работы различных приложений. Высокое значение имеет время реакции, оно требует надлежащей организации корпоративной сети и приложений. Работа в реальном времени стала жизненной необходимостью и одним из главных требований, предъявляемых к корпоративным сетям и приложениям.

Но при этом гарантировать хорошее время реакции особенно трудно - этому препятствует разнообразие потоков данных и их высокая интенсивность, потребность совершать поиск данных в базах большого объема, невысокая скорость глобальных линий связи между подразделениями, замедление скорости взаимодействия в шлюзах, согласующих неоднородные компоненты разных подсетей.

Поддержание работы учреждений, пользующихся данной сетью – одна из главных целей корпоративной сети. Пользователями корпоративной сети являются сотрудники данного предприятия, т.е. медицинские работники. Такая сеть включает в себя отделения в различных регионах по всей России, являясь составной сетью, включающей как глобальные, так и локальные сети.

Таким образом, актуальность темы дипломной работы обусловлена необходимостью создания надёжной и полнофункциональной корпоративной сети для медицинских учреждений.

Цель дипломной работы - разработка структуры корпоративной сети медицинских учреждений, расположенных в разных регионах Российской Федерации.

Для реализации этой цели были поставлены следующие задачи:

- Описать актуальность исследуемой проблемы.
- Рассмотреть «Единую регистратуру Новосибирской области».
- Провести анализ роли корпоративных сетей в сфере IT.

- Изучить особенности построения корпоративных сетей.
- Разработать физическую схему сети.
- Разработать пользовательское приложение для работы с рецептами.

1. Описание проблемы и предметной области

1.1 Актуальность проблемы.

Одной из видимых тенденций последнего времени стало развитие сетей, охватывающих несколько медицинских учреждений. Причем эти изменения заметны не только в крупных городах, но и в регионах. В настоящее время появляется все больше корпоративных сетевых клиник и частные медицинские сети, не связанные напрямую с корпорациями.

Под словосочетанием «сеть медицинских организаций» имеется в виду группа организаций, которая, независимо от организационно-правовой формы отдельных входящих в нее единиц или группы в целом и независимо от деталей системы управления, объединена наличием общих организационных, материальных и технологических ресурсов [4]. Иными словами, внутри сети происходит довольно свободный обмен информацией, средствами, кадрами. В государственных учреждениях элементы сетевой структуры могут быть представлены в виде административных и нормативно-методических связей между организационными единицами.

Для «сети медицинских организаций» необходимо создание корпоративной сети. Структура, определяющая характеристики сети, является важным параметром корпоративной сети. Именно поэтому структура сети может рассматриваться как объект управления, воздействие на который позволяет управлять потоками данных, что является основной задачей управления сетью.

В настоящее время разработано и используется огромное количество мощных систем управления корпоративными сетями. Это позволяет изучить результаты их работы и подчеркнуть общие для них положительные и отрицательные стороны. Связано это с необходимостью учитывать особенность работы корпоративной системы, что требует соответствующих методов управления ее работой и настроек корпоративной сети. Таким образом, постоянно углубляется разрыв между возрастающими возможностями систем управления и реальными потребностями при управлении, направленным на конкретные приложения.

В следствие этого разрабатываются новые концептуальные подходы к управлению корпоративными сетями. Они направлены на решение необходимого набора прикладных задач, и при применении универсальных многофункциональных систем управления обеспечивают требуемое качество их решения. Решение проблемы составлено на разработке подходов к управлению корпоративной сетью, совмещающих учет специфики решаемых задач и возможности действующих систем управления.

Исследования, направленные на создание и теоретическое обоснование указанных подходов актуальны как в настоящее время, так в перспективе развития корпоративных сетей.

1.2 Современное положение

До недавнего времени в российском здравоохранении почти полностью отсутствовали хоть какие-то признаки автоматизации. Карты, процедурные отчеты, бюллетени, учет лекарственных препаратов, пациентов – весь документооборот производился на бумаге. Это сказывалось на скорости, и стало быть на качестве обслуживания пациентов, также затрудняло работу медицинского персонала, вело к врачебным ошибкам, большим затратам времени на заполнение карт, составление отчетов. Отсутствие контроля работы подразделений, недостаток оперативной, аналитической информации осложняло руководство лечебно-профилактических учреждений (ЛПУ) и работу контролирующих органов.

В настоящее время в России достигнут значительный прогресс в обеспечении медицины электронными сервисами. Но предстоит еще долгий путь в направлении комплексной информатизации регионального здравоохранения.

Характерными особенностями развития информатизации здравоохранения в большинстве регионов на нынешнем этапе являются направленность на решение локальных учетно-расчетных задач и ведение документации, обеспечение делопроизводства. Во-вторых, не развиты аналитические схемы поддержки принятия решений и контроля использования ресурсов. В-третьих, отсутствие электронного документооборота и слабо развитый парк современного компьютерного оборудования. В-четвертых, несовершенное программное обеспечение, отсутствие лицензионного ПО, отсутствие защиты информации. В-пятых, проблема привлечения квалифицированных специалистов и низкий уровень компьютерной грамотности. [8]

Очевидно, что необходимо создание интегрированных информационных ресурсов системы регионального здравоохранения, единых решений по защите персональных данных и обеспечению систем безопасности, организации межведомственного взаимодействия с использованием медицинской информации. Развитие корпоративных порталов и всеобщее внедрение в ЛПУ Интернета способствует увеличению роли информационно-коммуникационных технологий в системе российского здравоохранения на уровне страны, и в масштабах отдельного региона.

1.3 «Электронная регистратура» Новосибирской области

Первые шаги в направлении информатизации лечебно-профилактических учреждений были сделаны в 2009 году. Все поликлиники города были дополнительно оснащены мини-АТС, компьютерами, справочными системами, телефонами, и системами для записи телефонных разговоров. Реализация «Городская электронная регистратура», основана на базе Дзержинской подстанции скорой помощи. Там организован единый call-центр, оборудованный современной техникой, позволяющей производить предварительную запись на прием к врачам городских поликлиник с использованием всех возможных средств связи.

Цель нововведения — повысить доступность медицинской помощи. Введение такой системы работы позволяет разгрузить и усовершенствовать регистрацию пациентов на прием.

Сайт же «Городская электронная регистратура Новосибирска» [6] был открыт осенью 2011 года. Подключить к ней все регионы планировалось до первого декабря 2012 года. В настоящее время Минздравом России осуществляется проект по созданию единой государственной информационной системы в здравоохранении на территории всей Российской Федерации. Важной частью этого проекта является сервис «Электронная регистратура» или «Запись на прием к врачу в электронном виде».

Во многих регионах есть самостоятельные наработки в этом направлении. Однако, единая система только создается. Первым же регионом, запустивший сервис в полном объеме, стала Омская область. В настоящее время 71 медицинская организация области может записывать пациента на прием к врачу через электронную систему, и все эти лечебно-профилактические учреждения интегрированы в федеральную систему.

2. Разработка структуры сети

2.1 Корпоративная сеть как объект исследования

Корпоративная сеть - это сложный комплекс взаимосвязанных и согласованно функционирующих аппаратных и программных компонентов, обеспечивающий передачу информации между различными удаленными приложениями и системами, используемыми на предприятии. По причине наличия нескольких центров обработки данных корпоративные сети принадлежат к децентрализованным (или распределенным) вычислительным системам.

Корпоративную сеть необходимо рассматривать с различных сторон: структурной, функциональной и системно-технической;

Со структурной точки зрения корпоративная сеть - сеть смешанной топологии, содержащая несколько локальных вычислительных сетей. Корпоративная сеть будет объединять филиалы ЛПУ, создавая общее информационное корпоративное пространство. С этой точки зрения корпоративная сеть отражает структуру медучреждения.

С функциональной точки зрения корпоративная сеть - это эффективная среда передачи актуальной информации необходимой для решения задач.



Рис. 1. Иерархия уровней корпоративной сети [9]

С системно-технической точки зрения корпоративная сеть представляет собой целостную структуру, состоящую из взаимосвязанных и взаимодействующих уровней, представленных на рис. 1.

Таким образом, с системно-технической точки зрения корпоративная сеть - это сложная система, предоставляющая пользователям и программам набор полезных в работе услуг и сервисов, общесистемных и специализированных приложений, обладающая набором полезных качеств и свойств, и содержащая в себе службы, гарантирующие нормальное функционирование корпоративной сети.

2.2 Роль корпоративных сетей в создании и развитии ИТ-инфраструктуры

В настоящее время ИТ-инфраструктура любого предприятия — это его ключевая инфраструктура, вне зависимости от вида деятельности. Для внедрения информационных технологий транспортную основу организуют корпоративные сети передачи данных.

Современная корпоративная сеть — это не только сеть передачи данных, а сложный комплекс, который способен предоставлять различные сервисы с прогнозируемыми характеристиками.

Благодаря корпоративным сетям результативно решаются задачи ключевых процессов. Таких как:

- быстрый доступ к информационным массивам общего информационного пространства;
- анализ состояния и управление бизнес-процессами из единого аналитического центра;
- обмен информационными и расчетными документами;
- непрерывное автоматизированное наблюдение (мониторинг) и управление ресурсами инфокоммуникационной системы из единого центра.

2.3 Особенности проектирования корпоративных сетей

Основная цель проектирования корпоративных сетей состоит в том, чтобы определить структуру, состав аппаратно-программных средств и организацию корпоративной сети. И при заданных ограничениях на затраты ее проектирования, внедрения и обслуживания они будут выполнять основные требования к качеству информационных услуг, предоставляемых сетью. И строится это на основании характеристик корпоративных информационных потоков предприятия, параметров

потребителей и производителей информации. При проектировании корпоративной сети сетевые администраторы и сетевые интеграторы стараются обеспечить выполнение следующих требований:[10]

- расширяемость - возможность простой интеграции отдельных компонентов сети (пользователей, приложений, служб, компьютеров).
- масштабируемость - возможность добавления новых узлов и протяженность связей, а также производительности узлов и сетевого оборудования;
- производительность - обеспечение необходимых значений параметров производительности сетевых узлов и каналов связи (скорость передачи данных, время реакции, задержка передачи и ее вариация);
- управляемость - обеспечение возможностей централизованного управления, планирования развития сети и мониторинга состояния сети;
- надежность - обеспечение бесперебойной работы узлов сети и каналов связи, согласованности, сохранности и доставки данных без изменений и ошибок узлу назначения;
- безопасность - обеспечение защиты данных от несанкционированного доступа.

Учитывая масштабность, использование глобальных связей, высокую степень разнородности проектирование корпоративных сетей является трудно формализуемым процессом. В сегодняшний день отсутствуют универсальные методики проектирования корпоративных сетей. Поэтому необходимо сформулировать некоторые типовые этапы выполнения сетевых проектов.

Процесс проектирования корпоративной сети состоит из следующих этапов: [2]

1. Анализ требований. На этом этапе формулируются основные цели предприятия (оперативный прием заказов, сокращение производственного цикла, повышение производительности труда и т.д.). Анализируются существующие аналогичные системы, обосновывается необходимость в собственных проектах системы.
2. Разработка бизнес-модели предприятия. Бизнес-модель или функциональная модель производства излагает основные, административные и вспомогательные бизнес-процессы предприятия, иерархические взаимоотношения и информационные потоки между подразделениями. Также передает структурированное отображение функций производственной системы, информации среды и объектов, связывающих эти функции.
3. Разработка технической модели корпоративной сети (структурный синтез). Техническая модель представляет собой совокупность технических средств,

необходимых для реализации проекта корпоративной сети. На данном этапе определяются технические параметры компонентов сети, такие как полный функциональный набор необходимых программных и аппаратных средств, но без конкретизации оборудования (марок и моделей).

4. Разработка физической модели корпоративной сети (параметрический синтез). Физическая же модель корпоративной сети представляет подробное описание программных и технических средств, их количества, технических параметров и способов взаимодействия. Таким образом, это конкретизацией технической модели сети, в которой выбраны протоколы, конкретные сетевые устройства и прочие сетевые технические средства. Выбираются они же в соответствии с техническими параметрами, задаваемыми в технической модели. Параметры, структурная схема и алгоритмы функционирования сети, как результаты выполнения данного этапа, используются для последующего анализа.
5. Моделирование и оптимизация корпоративной сети. Моделирование производится на данном этапе с целью оценки характеристик функционирования корпоративной сети и их оптимизации.
6. Установка и наладка корпоративной сети. На этом этапе подразумевается управление конфигурированием, координирование поставок от субподрядчиков, инсталляцию и наладку оборудования, обучение персонала.
7. Тестирование корпоративной сети. На этом этапе должны проводиться необходимые испытания, описанные в контракте с интегратором.
8. Сопровождение и эксплуатация корпоративной сети. Последний этап не имеет четко определенных временных границ, он предполагает непрерывный процесс.

2.4 Параметры качества корпоративной сети

В настоящее время постоянно растущие требования корпоративных пользователей и корпоративных приложений к пропускной способности сети привели к появлению новых высокоскоростных технологий и новых механизмов качества обслуживания, учитывающих различные характеристики трафика: относительная скорость передачи данных и чувствительность к задержкам, потерям и искажением пакетов. Рассмотрим основные параметры качества корпоративной сети: [2]

- пропускная способность сети - интегральный параметр характеризует объем информации, передаваемой сетью в единицу времени;
- реакция на характеристики профиля трафика — параметр, характеризующий изменения нагрузки на сеть в зависимости от характеристик профиля трафика.

Например, изменение числа искаженных или потерянных пакетов, пропускной способности при пульсирующем или плавном изменении трафика; количество искаженных или потерянных пакетов (согласно экспертным оценкам, для протокола TCP 1-5% потерянных пакетов находится в пределах нормы, предельное значение при котором сеть практически не работает - 40% потерянных или искаженных пакетов); [9]

- время доставки - время двойного хода (в прямом и обратном направлении). Этот параметр может изменяться в диапазоне от 0 до 2000 мс, оказывая влияние на производительность работы одного потока; [9]
- неравномерность времени доставки пакетов - параметр, влияющий на работу отдельных приложений, например, приложения, управляющие техническим объектом в реальном времени или передающие мультимедийную информацию.

2.5 Трехуровневая иерархическая модель

При проектировании корпоративной сети весь процесс разработки был разбит на три части, т.к. компьютерные сети удобно представлять в виде трехуровневой иерархической модели, которая содержит следующие уровни:

- уровень ядра;
- уровень распределения;
- уровень доступа.

Уровень ядра предназначен для высокоскоростной передачи сетевого трафика и скоростной коммутации пакетов. Поэтому на сетевых устройствах этого уровня не вводятся дополнительные технологии (списки доступа или маршрутизация по правилам), отвечающие за маршрутизацию и фильтрацию пакетов.

Уровень ядра или базовый уровень представим несколькими поликлиниками, расположенными в разных городах. Маршрутизаторы этих узлов – маршрутизаторы ядра – соединены между собой, образуя кольцевое ядро сети с избыточными путями. Этот уровень предназначен для оперативной и надежной коммутации больших объемов трафика. На базовом уровне трафик передается совместно для нескольких пользователей. Здесь обрабатываются большие объемы трафика, поэтому не менее важно учитывать скорость и задержки. Обычно используются быстродействующие сети Multi-Gigabit Ethernet и Gigabit Ethernet.

С помощью протокола Ethernet к каждому из маршрутизаторов подключается через коммутатор маршрутизатор и группа серверов, которые вместе образуют

демилитаризованную зону и обеспечивают доступ в Интернет. Группа корпоративных серверов подключается также к каждому узловому маршрутизатору. Каждый из маршрутизаторов ядра с помощью технологии глобальных сетей, например Frame Relay, соединен виртуальными каналами с маршрутизаторами районных поликлиник. Frame Relay ("Передача кадров") — технология передачи данных, активно применяющаяся в корпоративных сетях различного масштаба. Основным принцип этой технологии состоит в создании нескольких виртуальных каналов на одном физическом, при этом для каждого виртуального канала резервируется гарантированная полоса пропускания. Этот принцип дает ряд существенных преимуществ и перед выделенными цифровыми каналами, и перед протоколами X.25 и TCP/IP.[9]

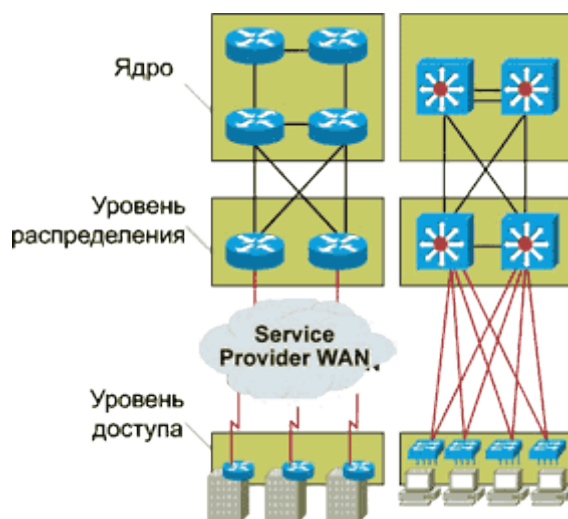


Рис. 2. Трехуровневая модель [11]

Уровень распределения используется для суммирования маршрутов. Суммирование проводится для уменьшения сетевого трафика на верхних уровнях сети. Оно представляет собой объединение нескольких сетей в одну большую общую. Основными функциями уровня распределения являются фильтрация, маршрутизация и доступе к региональным сетям. Если необходимо, то и определение правил доступа пакетов к уровню ядра. На уровне распределения необходимо устанавливать наиболее быстрый способ обработки запросов к службам, такой как метод файлового обращения к серверу. Маршрутизаторы уровня распределения соединены с маршрутизаторами ядра. На уровне доступа производится контроль доступа к сети и формируется сетевой трафик. В основном используются сети 100-Mbps Fast Ethernet и 1000-Mbps Gigabit Ethernet. [11]

Маршрутизаторы уровня доступа служат для подключения к глобальной вычислительной сети отдельных пользователей (серверы доступа) или отдельных локальных сетей. На этом уровне реализовано управление пользователями и рабочими

группами при обращении к ресурсам объединенной сети. Иногда уровень доступа называют уровнем настольных систем. Наибольшая часть необходимых пользователям сетевых ресурсов должна быть доступна локально. На уровне распределения выполняется перенаправление трафика к удаленным службам. Скорость сети - Ethernet 10 Mbps или 100-Mbps Fast Ethernet. [11]

Самым простым коммутирующим оборудованием уровня доступа являются коммутаторы рабочих групп. В свою очередь, к ним присоединяются автоматизированные рабочие места сотрудников организации (АРМы). По причине большого количества АРМов в сети коммутаторы уровня рабочих групп необходимо разделить на два уровня. Коммутаторы рабочих групп верхнего (второго) уровня объединяются в единую сеть с помощью коммутаторов зданий, которые в рамках одного кампуса соединяются в кольцо оптоволоконными линиями связи.

Проектируемая сеть должна соответствовать требованиям избыточности и структурированности. Избыточность делает сеть устойчивой к нарушениям каналов передачи данных и их неполадкам, повышает надежность системы, однако и увеличивает трудоемкость администрирования сети.

2.6 Структура сети

Территориальные компьютерные сети служат для обмена данными между поликлиниками, расположенными в разных регионах страны. Они обладают большой протяженностью и требуют больших затрат. В их стоимость входят кабели, работа по прокладке, затраты на коммутационное оборудование, промежуточную усилительную аппаратуру, обеспечивающую необходимую полосу пропускания канала и эксплуатационные затраты для поддержания в рабочем состоянии.

Глобальная сеть не может быть полностью создана для медицинских учреждений, поэтому предлагается промежуточный вариант: корпоративная сеть медицинских учреждений использует оборудование общественной глобальной сети, их услуги, но часть дополняет своими собственными. Например, аренда каналов связи, на основе которой создать собственную территориальную сеть.

Локальные и глобальные сети состоят из периферийных подсетей и магистрали, которая связывает эти подсети. Структура крупной локальной сети приведена на рис. 3, она состоит из подсетей, объединенных магистралью, включающих два кольца FDDI и четыре маршрутизатора. Каждая подсеть также может иметь иерархическую структуру, образованную своими маршрутизаторами, коммутаторами, концентраторами и сетевыми

адаптерами. Все эти коммуникационные устройства связаны разветвленной кабельной системой. Такая сеть может быть расположена на территории нескольких районов города.

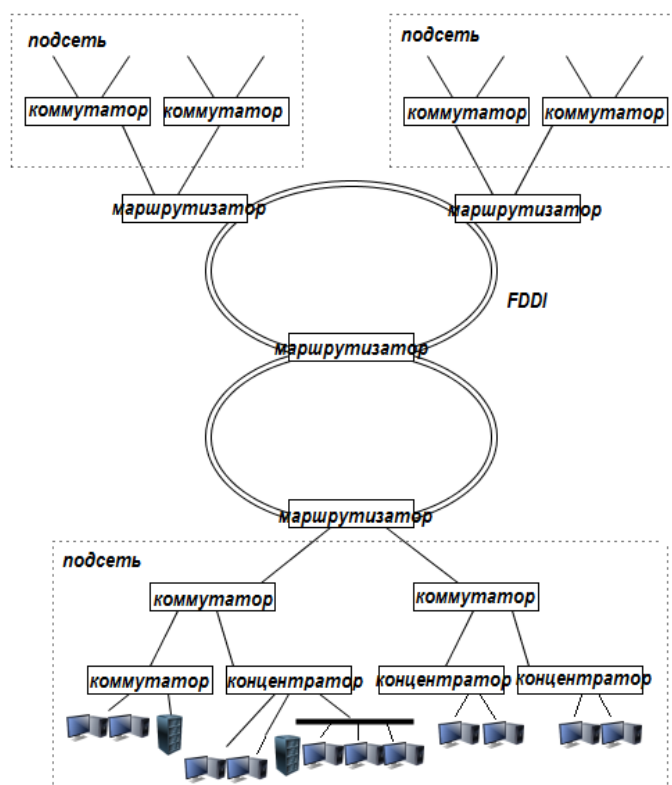


Рис. 3. Структура локальной сети

Глобальная сеть соединяет локальные сети, расположенные на большом расстоянии друг от друга. Она имеет иерархическую структуру с высокоскоростной магистралью, более медленными периферийными сетями и каналами доступа локальных сетей к глобальным.

Структура глобальной компьютерной сети приведена на рис. 4. В основе лежат некоммутируемые каналы связи, которые соединяют коммутаторы глобальной сети. Коммутаторы называют также центрами коммутации пакетов (ЦКП).

В местах, где требуется ответвление или соединение потоков данных или магистральных сетей устанавливаются коммутаторы. Выбор такого месторасположения коммутаторов определяется также возможностью обслуживания коммутаторов квалифицированным персоналом, наличием выделенных каналов связи в данном пункте, надежностью сети, определяемой избыточными связями между коммутаторами.

Конечные узлы глобальной сети показаны на рис. 4. Основные типы конечных узлов глобальной сети: компьютеры (К), локальные сети, маршрутизаторы (R), мультиплексоры (MUX), , используются для одновременной передачи по компьютерной сети данных и

голоса (или изображения). Устройства типа DTE (Data Terminal Equipment) преобразуют пользовательскую информацию в данные для передачи по линии связи и осуществляющее обратное преобразование. Локальная сеть отделена от глобальной маршрутизатором или удаленным мостом.

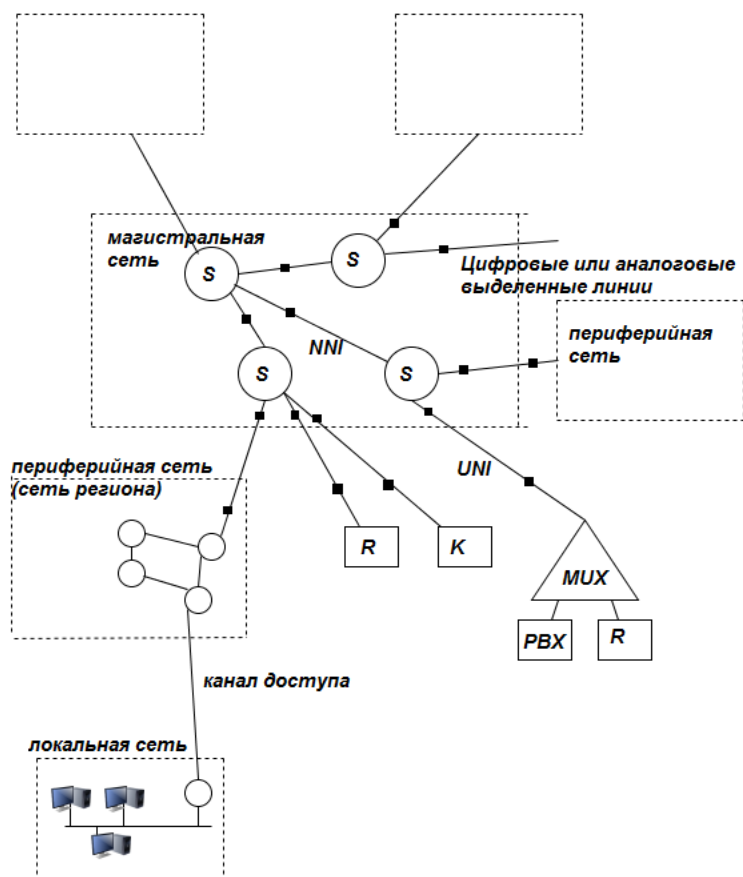


Рис. 4. Структура глобальной сети

Здесь используются следующие обозначения: *S* (switch) - коммутаторы, *K* - компьютеры, *R* (router) - маршрутизаторы, *MUX* (multiplexor) - мультиплексор, *UNI* (User-Network Interface) – «интерфейс пользователь – сеть» и *NNI* (Network-Network Interface) – «интерфейс сеть – сеть». Офисная АТС - *PBX*, черные квадратики - устройства *DCE*.

При передаче данных через глобальную сеть мосты и маршрутизаторы работают в соответствии с той же логикой, что и при соединении локальных сетей. Мосты строят таблицу MAC-адресов на основании проходящего трафика, и по информации в этой таблице принимают решение - предоставлять кадры в удаленную сеть или нет.

Маршрутизаторы принимают решение на основании номера сети пакета какого-либо протокола сетевого уровня (например, IP или IPX). Если же пакет нужно передать следующему маршрутизатору по глобальной сети упаковывают его в кадр этой сети, дополняют соответствующим аппаратным адресом следующего маршрутизатора и посылают в глобальную сеть.

Так как конечные узлы глобальной сети должны передавать данные по каналу связи конкретного стандарта, то каждое устройство типа DTE необходимо оснастить устройством типа DCE (Data Circuit terminating Equipment) которое обеспечивает необходимый протокол физического уровня заданного канала. В зависимости от типа канала для связи с каналами глобальных сетей используются DCE трех основных типов: модемы для работы по выделенным и коммутируемым аналоговым каналам, устройства DSU/CSU для работы по цифровым выделенным каналам сетей технологии TDM и терминальные адаптеры (ТА) для работы по цифровым каналам сетей ISDN. Устройства DTE и DCE обобщенно называют оборудованием, размещаемым на территории абонента глобальной сети - Customer Premises Equipment, CPE.

При использовании общественной сети очень важны предоставляемые сетью услуги и правильное определение интерфейса взаимодействия с сетью. Оконечное оборудование и программное обеспечение должны корректно сопрягаться с соответствующим оборудованием и программным обеспечением общественной сети, поэтому в глобальной сети строго прописан и стандартизован интерфейс «пользователь-сеть» (UNI - User-to-Network Interface). Это необходимо для того, чтобы пользователи могли без проблем подключаться к сети с помощью коммуникационного оборудования любого производителя, который соблюдает стандарт UNI данной технологии (например, X.25).

Протоколы взаимодействия коммутаторов внутри глобальной сети, называемые интерфейсом «сеть-сеть» (Network-to-Network Interface, NNI), стандартизуются не всегда. Если организация создает глобальную сеть, она должна иметь свободу действий, чтобы решать, как взаимодействовать внутренним узлам сети между собой. Поэтому внутренний интерфейс, в случае его стандартизации, носит название «сеть-сеть», а не «коммутатор-коммутатор», подчеркивая, что он должен использоваться в основном при взаимодействии двух территориальных сетей разных операторов. Но если стандарт NNI принимается, то в соответствии с ним организуется взаимодействие всех коммутаторов сети, а не только пограничных.

3. Приложение для работы с рецептами

Человеку свойственно забывать, и каждый из нас не раз уже страдал от этого. Вопрос напоминания о некоторых значимых событиях жизни всегда стоял довольно остро, особенно о приеме некоторых лекарств и их дозировке.

Практической целью данной работы было создание приложения для работы с рецептами, которое будет напоминать пациенту о приеме лекарств.

Рецепт – это письменное распоряжение врача аптеке о приготовлении и отпуске лекарств, которое также содержит указания, как ими пользоваться. Рецепт составляют по определённой форме и правилам. Простой рецепт — рецепт, выписанный на одно лекарственное средство, комплексный — если в рецепте присутствуют несколько лекарственных препаратов. Все рецепты в данном приложении разбиваются на простые.

Каждый рецепт включает в себя:

- Название лечебно-профилактического учреждения (ЛПУ), адрес и телефон.
- Номер рецепта.
- Дата выдачи рецепта.
- Ф.И.О. пациента, возраст,
- Адрес прописки,
- Номер истории болезни.
- Ф.И.О. врача, выписавшего рецепт.
- Лекарственные препараты.
- Режим и длительность приема.
- Срок годности рецепта (указывает врач).

Перед тем как перейти к разработке приложения, было проанализированы и составлены требования. Были сформированы следующие предполагаемые функции:

- Просмотр выписанных рецептов как активных, так и неактивных.

Рецепт считается активным, если на сегодняшний день приходится прием лекарств. Как только заканчивается цикл приема препаратов, рецепт становится не активным.

- Установка (удаление) напоминания на электронную почту.

Необходима регистрация электронного ящика в системе.

- Поиск обычный и расширенный. Поиск по дате выдачи рецепта, по специальности врача, по препарату, по примечанию.

- Просмотр статистики посещений врачей по месяцам, годам. Количество посещений врачей по специальности.

3.1 Средства разработки

Для реализации приложения был выбран язык программирования Java. Преимущества данного языка заключаются в том, что существуют различные фреймворки, создать структуру веб-сайта в рамках одного проекта, используя единую среду разработки. В качестве среды разработки была выбрана IntelliJ IDEA, которая обладает интегрированной поддержкой различных систем контроля версий, фреймворков, которые были использованы при написании программы к дипломной работе.

Также была использована технология JSP, которая позволяет создавать содержимое, имеющее статические и динамические компоненты.

Преимущества технологии JSP заключается в том, что JSP является платформонезависимой, переносимой и легко расширяемой технологией для разработки веб-приложений. Также JSP отличается высокой производительностью, потому что весь код страницы транслируется в java-код сервлета с помощью компилятора JSP страниц Jasper, и затем компилируется в байт-код виртуальной машины java (JVM).

Сервер приложений был выбран GlassFish благодаря высокой надежности, доступности, полной поддержке промышленной кластеризации и широкому спектру функций.

Для проектирования и описания UML диаграмм использовался пакет Astah/JUDE.

Для автоматизации сборки проектов, специфицированных на XML-языке POM использован фреймворк Apache Maven, который обеспечивает декларативную, а не императивную сборку проекта. То есть, в файлах проекта pom.xml содержится его декларативное описание, а не отдельные команды. Все задачи по обработке файлов Maven выполняет через плагины.

Настройки Maven для работы с Glassfish представлены в приложении А.

Настройки Maven для работы с внешними модулями и библиотеками представлены в приложении Б.

3.2 Реализация основной функциональности

Основные функции данного приложения представлены на рисунке 5.

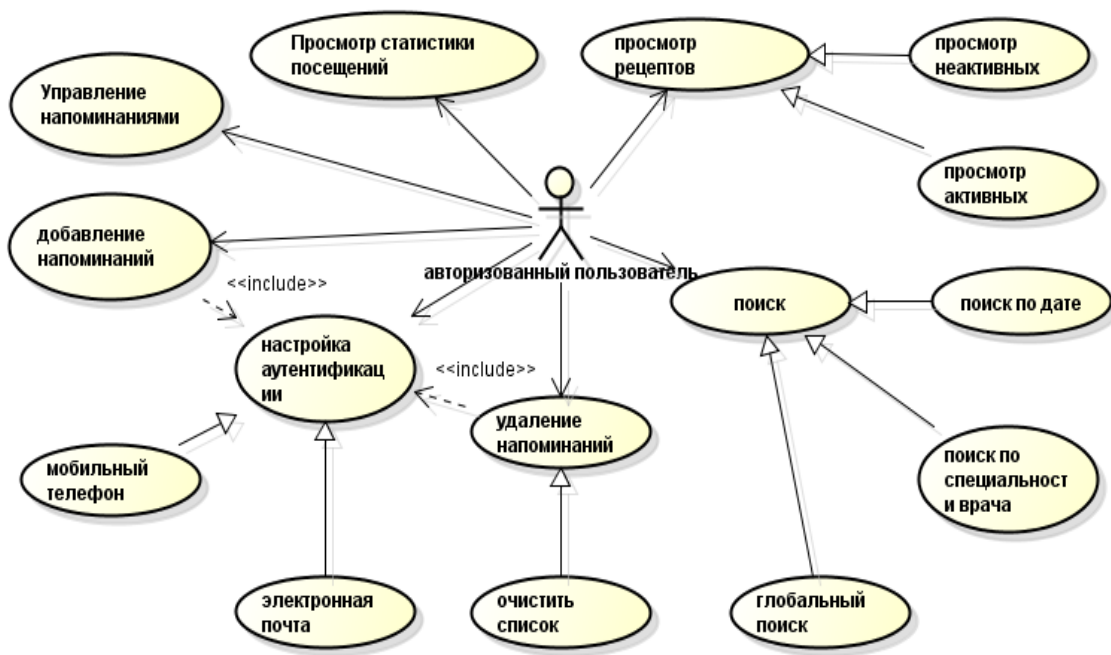


Рисунок 5. Use-case Model. Основные функции приложения

- Просмотр выписанных рецептов (рис. 6).

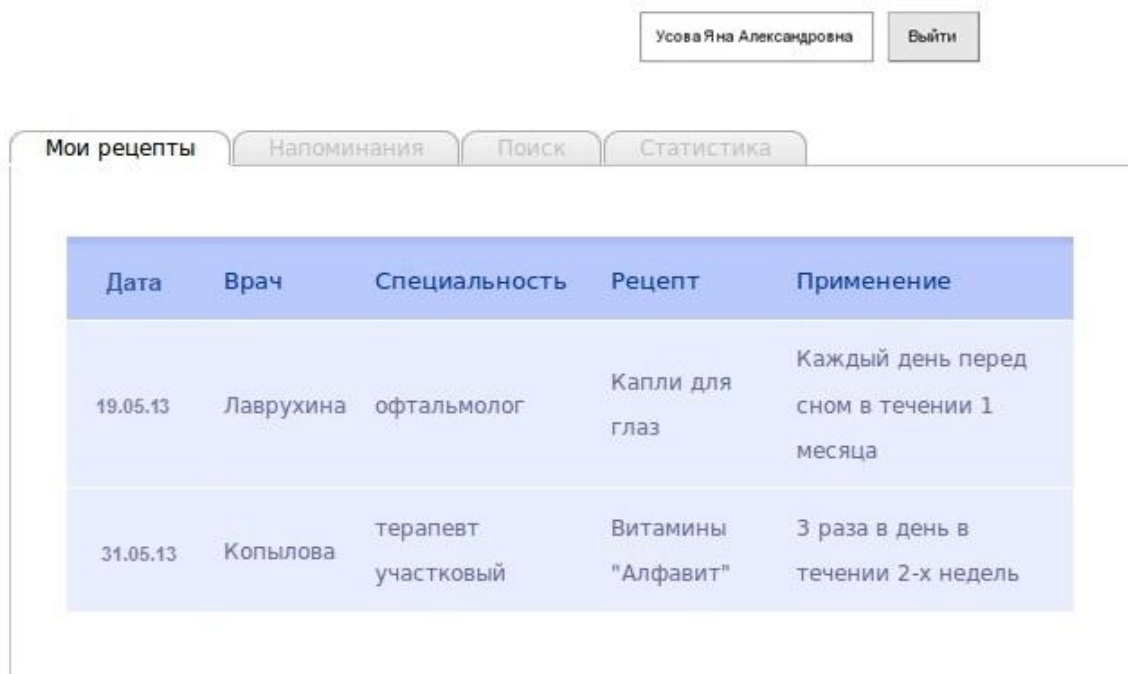


Рисунок 6. Просмотр выписанных рецептов

Одной из главных функций приложения является просмотр выписанных рецептов как активных, так и неактивных. Рецепт включает в себя дату, специальность врача, фамилию, имя, отчество, выписанные препараты, режим и длительность приема, название лечебно-профилактического учреждения (ЛПУ), адрес и телефон, примечание. Окно с активными рецептами открывается авторизованным пользователям сразу после открытия приложения. Для просмотра неактивных рецептов необходимо нажать на чекбокс внизу страницы.

- Установка напоминаний о приеме лекарств (рис. 7).

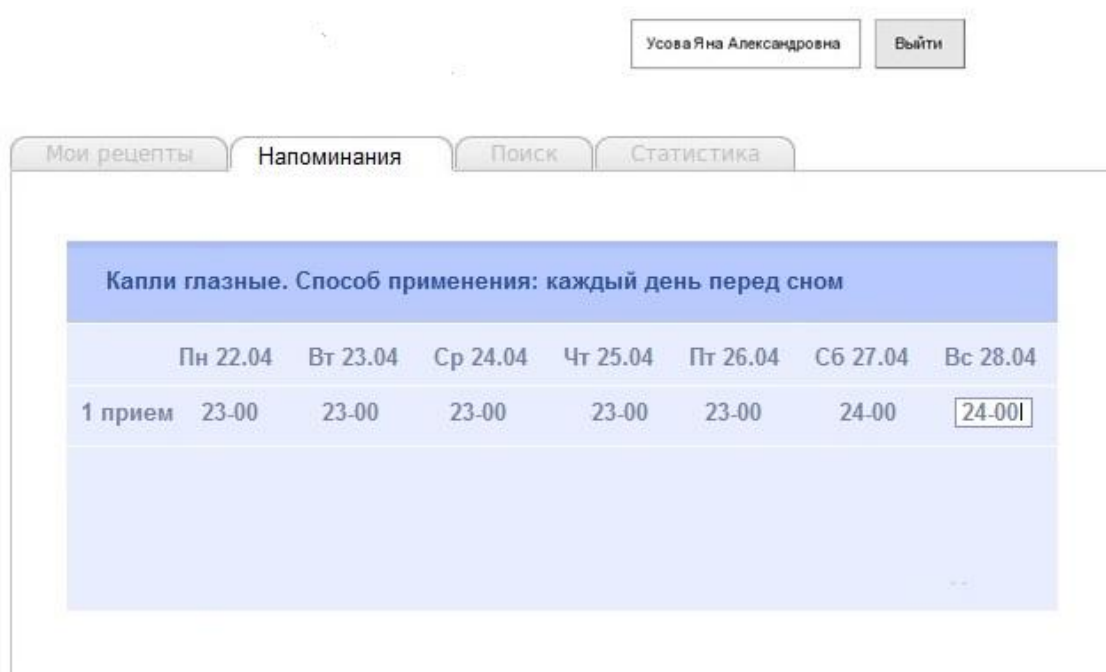


Рис. 7. Установка напоминаний о приеме лекарств

Для использования этой функции необходимо указать электронную почту. Для каждого лекарства из активного рецепта представлена своя таблица, где количество строк равно количеству приема в день, столбцы – дни недели, значения в ячейках – время приема. Не обязательно указывать каждое время приема. По умолчанию время для завтрака 8-00, обед в 14-00, ужин в 19-00. Особое внимание уделено разработке интуитивно понятного интерфейса для установки времени и даты напоминания не требующего никаких технических навыков и знаний английского языка.

- Удаление напоминаний.

Возможность удаления уже установленных напоминаний.

- Поиск по специальности, фамилии врача, дате. Расширенный поиск (рис. 8).

The screenshot displays the 'Поиск' (Search) tab of a medical application. At the top right, there is a user profile section with the name 'Усова Яна Александровна' and a 'Выйти' (Logout) button. Below this, a navigation bar contains four tabs: 'Мои рецепты', 'Напоминания', 'Поиск', and 'Статистика'. The 'Поиск' tab is active. It features two search modes: 'Обычный поиск' (Simple search) and 'Расширенный поиск' (Advanced search). The 'Обычный поиск' mode has a single text input field and a 'Поиск' button. The 'Расширенный поиск' mode has three input fields labeled 'Дата', 'Врач', and 'Название', and a 'Поиск' button.

Рис. 8. Расширенный поиск

Поиск можно производить из вкладки «поиск», либо же с главной страницы приложения. Поиск по дате производится в режиме календаря. По умолчанию отмечена текущая дата. Прошедшие даты неактивны и обозначены серым цветом. Перейти на другой месяц можно прокруткой.

- Просмотр статистики (рис. 9).

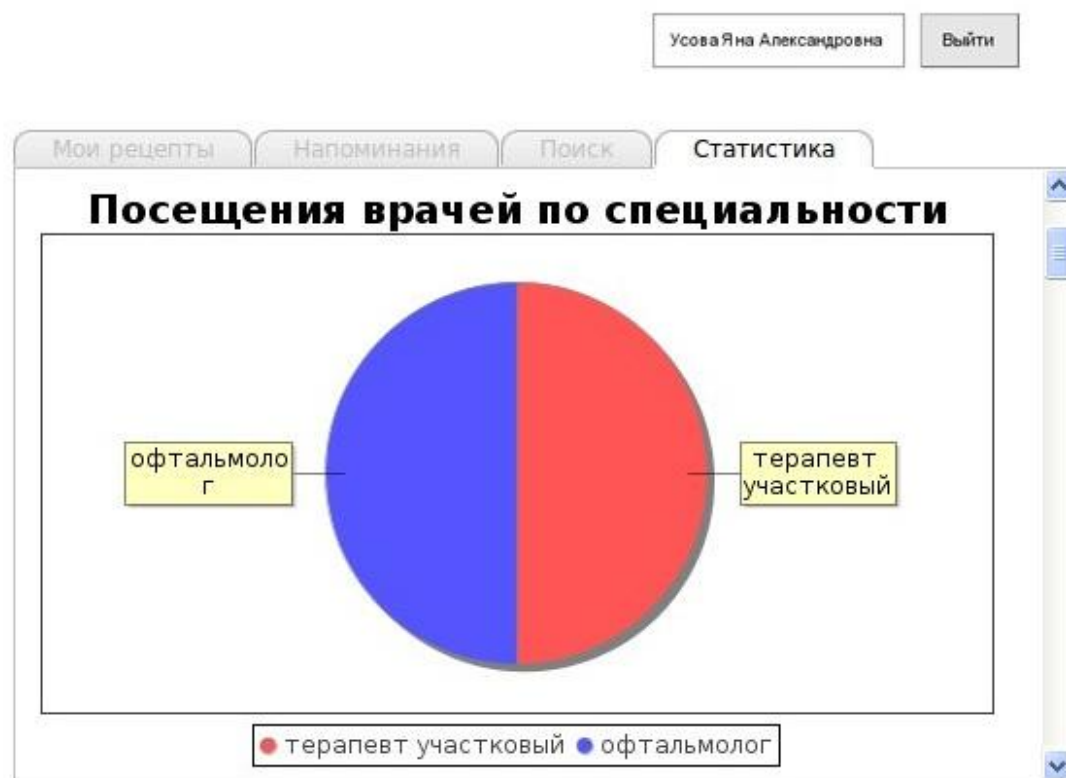


Рисунок 9. Просмотр статистики

Возможность просмотра статистики и визуализация данных открывает широкие возможности для анализа собственного состояния, определение сезонности заболевания (аллергические реакции, астма), характерной продолжительности, получении информации о лекарственных препаратах, принимаемых в период протекания заболеваний.

Статистические данные: дозировка лекарственных препаратов, частота посещений врачей.

Заключение

Основой инфраструктуры современных предприятий являются корпоративные сети передачи данных, обеспечивающие транспорт для переноса информации между различными приложениями информационных систем.

Основными задачами корпоративной сети являются:

- оперативный доступ к информационным массивам в рамках единого информационного пространства;
- анализ состояния и управление бизнес-процессами из единого аналитического центра;
- обмен расчетными, информационными документами;
- постоянный автоматизированный мониторинг и управление ресурсами инфокоммуникационной системы из единого центра.

В ходе дипломной работы была рассмотрена «Единая регистратура Новосибирской области», проведен анализ роли корпоративных сетей в сфере IT, разработана структура сети, позволяющая объединить поликлиники разных городов и регионов. Определена топология сети, содержащая в себе уровень ядра и уровень доступа.

Практической частью была разработка приложения, позволяющего пользователям работать с выписанными рецептами, устанавливать напоминания о приеме лекарственных средств. Основными функциями приложения являются:

- Просмотр выписанных рецептов как активных, так и неактивных.
- Установка (удаление) напоминания на электронную почту.
- Поиск обычный и расширенный. Поиск по дате выдачи рецепта, по специальности врача, по препарату, по примечанию.
- Просмотр статистики посещений врачей по месяцам, годам. Количество посещений врачей по специальности.

Для реализации данного приложения были выбраны следующие программные средства и технологии: объектно-ориентированный язык программирования Java, технология JavaServer Pages, сервер приложений Oracle GlassFish Server.

Список литературы

1. Аверьянов А.Г., Анисимов А.И. и др. Многоуровневые информационно-управляющие системы реального времени для топливно-энергетического комплекса России: монография / под ред. Костюкова В.Е. - Нижний Новгород: Изд-во Нижегородского университета, 2007. - 243 с.
2. Альтман Е. А. Проектирование корпоративной сети: методические указания. Е. А. Альтман. Омск: Омский государственный университет путей сообщения, 2004. 28 с.
3. Бертсекас Д., Галлагер Р. Сети передачи данных. : Пер. с англ. М.: Мир, 1989. 544 с.
4. Борейко А.А., Орлинский Д.Б. Особенности автоматизации сети медицинских организаций – проблемы и решения // Главный врач: хозяйство и право – 2008 - №5, С. 38 – 41.
5. Вишнеков А.В., Леохин Ю.Л. Оценка надежности управляющих программ реального времени // Научно-техническая конференция «МИКРОСИСТЕМА - 93». Тезисы докладов. - Москва, 1993. - С. 21-22.
6. Единая регистратура Новосибирской области [Электронный ресурс]. - Режим доступа: <http://www.zdravnsk.ru>. Дата обращения: 10.04.2013.
7. Иванов И.А., Леохин Ю.Л. Формирование прогноза состояния компьютерной сети // Свидетельство об официальной регистрации программы для ЭВМ № 2005610845 от 11.04.05. - Москва. - Федеральная служба по интеллектуальной собственности, патентам и товарным знакам.
8. Ильницкий А. Н., Чурносков М. И. Внедрение системы дистанционной записи пациентов на консультационный прием в областную консультативную поликлинику как средство оптимизации медицинских технологий [Электронный ресурс]. – режим доступа: <http://catalog-statei.ru/scientific-articles.php?id=9>. Дата обращения: 12.03.2013 г.
9. Леохин Ю.Л. Научные основы управления параметрами структур корпоративных сетей: дис. док. техн. наук Моск. гос. института электроники и математики, Москва, 2009
10. Олифер В.Г. Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов. 3-е изд. / В. Г. Олифер, Н. А. Олифер. – СПб.: Питер, 2008. – 958 с.
11. Enterprise Campus 3.0 Architecture: Overview and Framework [Электронный ресурс]. – Режим доступа: <http://www.cisco.com/en/US/docs/solutions/Enterprise/Campus/Campover.html>. Дата обращения: 5.05.2013 г.

12. Goncalves Antonio. Beginning Java EE 6 with GlassFish 3. 2-е изд. Goncalves Antonio – Apress, 2010. – 537 с.

Приложение А

Настройки Maven для работы с Glassfish.

Файл /home/usova/settings.xml.

```
<?xml version="1.0" encoding="UTF-8"?>
<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0
  http://maven.apache.org/xsd/settings-1.0.0.xsd">
  <profiles>
    <profile>
      <id>glassfish-context</id>
      <properties>
        <local.glassfish.home>/home/glassfish</local.glassfish.home>
        <local.glassfish.user>admin</local.glassfish.user>
        <local.glassfish.domain>domain1</local.glassfish.domain>

        <local.glassfish.passfile>${local.glassfish.home}/glassfish/domains/${local.glassfish.domain}/master-
        password</local.glassfish.passfile>
      </properties>
    </profile>
  </profiles>

  <activeProfiles>
    <activeProfile>glassfish-context</activeProfile>
  </activeProfiles>
</settings>
```

Часть файла /home/usova/pom.xml.

```
<profiles>
  <profile>
    <id>glassfish-deploy</id>
    <pluginRepositories>
      <pluginRepository>
        <id>maven.java.net</id>
        <name>Java.net Maven2 Repository</name>
        <url>http://download.java.net/maven/2</url>
      </pluginRepository>
```

```

</pluginRepositories>
<build>
  <plugins>
    <plugin>
      <groupId>org.glassfish.maven.plugin</groupId>
      <artifactId>maven-glassfish-plugin</artifactId>
      <version>2.1</version>
      <configuration>
        <glassfishDirectory>${local.glassfish.home}</glassfishDirectory>
        <user>${local.glassfish.user}</user>
        <passwordFile>${local.glassfish.passfile}</passwordFile>
        <autoCreate>true</autoCreate>
        <debug>true</debug>
        <echo>false</echo>
        <terse>true</terse>
        <domain>
          <name>${local.glassfish.domain}</name>
          <adminPort>4848</adminPort>
          <httpPort>8080</httpPort>
          <httpsPort>8443</httpsPort>
          <iiopPort>3700</iiopPort>
          <jmsPort>7676</jmsPort>
          <reuse>false</reuse>
        </domain>
        <components>
          <component>
            <name>${project.artifactId}</name>
            <artifact>
              ${project.build.directory}/${project.build.finalName}.war
            </artifact>
          </component>
        </components>
      </configuration>
    </plugin>
  </plugins>
</build>
</profile>
</profiles>

```

Приложение Б

Настройки Maven для работы с внешними модулями и библиотеками.

Часть файла `home/usova/pom.xml`.

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-
v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.ccfite.usova.recipes</groupId>
  <artifactId>JSPServlet</artifactId>
  <packaging>pom</packaging>
  <version>1.0-SNAPSHOT</version>
  <name>Parent</name>
  <url>http://maven.apache.org</url>

  <dependencies>
    <dependency>
      <groupId>log4j</groupId>
      <artifactId>log4j</artifactId>
    </dependency>
    <dependency>
      <groupId>org.hibernate</groupId>
      <artifactId>hibernate-core</artifactId>
      <version>4.2.1.Final</version>
    </dependency>
    <dependency>
      <groupId>jstl</groupId>
      <artifactId>jstl</artifactId>
      <version>1.2</version>
    </dependency>
    <dependency>
      <groupId>javax.servlet</groupId>
      <artifactId>jsp-api</artifactId>
      <version>2.0</version>
      <scope>provided</scope>
```

```
</dependency>
<dependency>
  <groupId>org.jfree</groupId>
  <artifactId>jfreechart</artifactId>
  <version>1.0.14</version>
</dependency>
</dependencies>
</project>
```

Приложение В.

Файл Doctor.java

```
package ru.ccfit.usova.model;
import javax.persistence.*;
import java.util.List;
@Entity
@Table(name = "Доктора")
public class Doctor {
    @Id
    @GeneratedValue(strategy = GenerationType.AUTO)
    private int id;

    @Column(name = "фамилия")
    private String surname;

    @Column(name = "имя")
    private String name;

    @Column(name = "отчество")
    private String patronimic;

    @Column(name = "специализация")
    private String specialization;

    @Column(name = "поликлиника")
    private String clinic;

    @OneToMany(mappedBy = "doctor", fetch = FetchType.EAGER, cascade =
CascadeType.ALL)
    private List<Recipe> recipes;
    public int getId() {
        return id;
    }
}
```

```

public void setId(int id) {
    this.id = id;
}
public String getSurname() {
    return surname;
}
public void setSurname(String surname) {
    this.surname = surname;
}
public String getName() {
    return name;
}
public void setName(String name) {
    this.name = name;
}
public String getPatronimic() {
    return patronimic;
}
public void setPatronimic(String patronimic) {
    this.patronimic = patronimic;
}
public String getSpecialization() {
    return specialization;
}
public void setSpecialization(String specialization) {
    this.specialization = specialization;
}
public String getClinic() {
    return clinic;
}
public void setClinic(String clinic) {
    this.clinic = clinic;
}
}

```


Приложение Г.

Файл Dao.java

```
package ru.ccf.it.usova.dao;

import org.hibernate.Query;
import org.hibernate.Session;
import org.hibernate.Transaction;
import ru.ccf.it.usova.model.Patient;
import ru.ccf.it.usova.util.HibernateUtil;

import java.util.ArrayList;
import java.util.List;

public class Dao {
    public void addPatient(Patient patient) {
        Transaction trns = null;
        Session session = HibernateUtil.getSessionFactory().openSession();
        try {
            trns = session.beginTransaction();
            session.save(patient);
            trns.commit();
        } catch (RuntimeException e) {
            if (trns != null) {
                trns.rollback();
            }
            e.printStackTrace();
        } finally {
            session.flush();
            session.close();
        }
    }

    public void updatePatient(Patient patient) {
        Transaction trns = null;
```

```

Session session = HibernateUtil.getSessionFactory().openSession();
try {
    trns = session.beginTransaction();
    session.update(patient);
    session.getTransaction().commit();
} catch (RuntimeException e) {
    if (trns != null) {
        trns.rollback();
    }
    e.printStackTrace();
} finally {
    session.flush();
    session.close();
}
}

public void deletePatient(int id) {
    Transaction trns = null;
    Session session = HibernateUtil.getSessionFactory().openSession();
    try {
        trns = session.beginTransaction();
        Patient user = (Patient) session.load(Patient.class, new Integer(id));
        session.delete(user);
        session.getTransaction().commit();
    } catch (RuntimeException e) {
        if (trns != null) {
            trns.rollback();
        }
        e.printStackTrace();
    } finally {
        session.flush();
        session.close();
    }
}
}

```

```

@SuppressWarnings("unchecked")
public List<Patient> getAllPatients() {
    List<Patient> users = new ArrayList<Patient>();
    Transaction trns = null;
    Session session = HibernateUtil.getSessionFactory().openSession();
    try {
        trns = session.beginTransaction();
        users = session.createQuery("from Patient").list();
    } catch (RuntimeException e) {
        e.printStackTrace();
    } finally {
        session.flush();
        session.close();
    }
    return users;
}

public Patient getPatientById(int userid) {
    Patient patient = null;
    Transaction trns = null;
    Session session = HibernateUtil.getSessionFactory().openSession();
    try {
        trns = session.beginTransaction();
        String queryString = "from Patient where id = :id";
        Query query = session.createQuery(queryString);
        query.setInteger("id", userid);
        patient = (Patient) query.uniqueResult();
    } catch (RuntimeException e) {
        e.printStackTrace();
    } finally {
        session.flush();
        session.close();
    }
    return patient;
}
}

```