Государственное образовательное учреждение высшего профессионального образования

|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | ***«Московский государственный технический университет  имени Н.Э. Баумана»***  ***(МГТУ им. Н.Э. Баумана)*** |

ФАКУЛЬТЕТ Биомедицинская техника

КАФЕДРА БМТ2 «Инженерное дело в медико-биологической практике»

**РАСЧЁТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**к курсовой работе на тему:**

«Разработка системы хранения данных рефлексодиагностических исследований»

по курсу «КТ в МБИ»

Студент группы БМТ2-92  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Дёшин И.А.**

(Подпись, дата)

Руководитель курсовой работы **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Косоруков А.Е.**

(Подпись, дата)

Москва, 2014

1. Введение

Система разрабатывалась для хранения биомедицинских данных для аппаратов акупунктурой диагностики. Данные приборы используются в диагностических целях для выявления отклонений в физиологическом состоянии человека. Именно на диагностику делается упор в современной медицине. Для примера рассмотрим нормативные объёмы медицинской помощи населению (см. таблицу 1).

*Таблица 1.*

| 1. медицинская помощь в амбулаторных условиях, оказываемая с профилактическими и иными целями | 2.98  (посещения на 1 жителя) |  |
| --- | --- | --- |
| 1. медицинская помощь в амбулаторных условиях, оказываемая в связи с заболеваниями | 2.18  (обращений на 1 жителя) |
| 1. медицинская помощь в амбулаторных условиях, оказываемая в неотложной форме | 0.6  (посещения на 1 жителя) |
| 1. скорая медицинская помощь вне медицинской организации | 0.318  (вызова на 1 жителя) | *Схема 1.* |
| Данные взяты из постановления правительства Российской Федерации от 18 октября 2013 г. N 932 г. Москва. | | |

Как видно из приведённых данных в основном медицинская помощь нацелена на диагностику, в меньшей степени на лечение, и около 17% на неотложную помощь. Так же в современной медицине делается акцент на сокращение второй группы за счёт увеличения объёма помощи в первой (Своевременное обнаружение отклонений и их корректировка).

Для этих целей хорошо подходят методики акупунктурной диагностики, так как они затрагивают множество групп органов и позволяют проводить не инвазивные экспресс обследования. Приборы для диагностики, соответственно измеряют один или несколько отличительных физических признаков точек акупунктуры (согласно Ф.Г. Портнову(1987)):

1. низкий порог чувствительности;
2. высокая локальная температура;
3. повышенное кожное дыхание;
4. низкое электрическое сопротивление при исследовании постоянным или переменным током (20-250 кОм);
5. большая электрическая емкость (0,1-1,0 мкФ);
6. высокий электрический потенциал (до 350 мВ).
7. Требования к ПО

Разрабатывалась система хранения данных для приборов измерения электрических характеристик точек акупунктуры. Поскольку съём производится с помощью импульсного электрического сигнала специфичной формы, имеется возможность хранения данных в двух видах:

1. Непосредственно данные электрического сигнала (напряжение и ток)
2. Данные рассчитанных характеристик схемы замещения БО.

Второй вид не смотря на большую информативность и компактность (Количество элементов в схеме замещения значительно ниже чем количество отсчётов в электрическом сигнале) имеет один существенный недостаток, а именно не возможность добавления новых показателей при модернизации методик расчёта или пересчёта старых при корректировке схемы. Поэтому приоритетней хранить не обработанные сырые данные для дальнейшего использования.

Поскольку одним из требований к разработке системы была открытость измеряемых данных для дальнейшей экспериментальной обработки, то для хранения электрических характеристик сигнала должен быть использован типизированный файл с записями мгновенных значений в текстовом виде. Со строчками такого типа: string str = “ ” + V.ToString() + “ ” + I.ToString(), где int V,I.

Для записей, данных о пациентах лучше всего использовать простой бинарный файл. Такой файл подлежит прочтению только с помощью программы и обеспечивает необходимый уровень защиты данных пациента. При необходимости можно придумать дополнительно алгоритм шифрования.

По мимо записей измерений и пациентов предполагаются записи об операторах. Для них можно использовать тот же формат хранения данных, что и для пациентов. Однако для дифференцировки пользователей должны быть различные права, в противном случае эти данные не несут смысла. Поэтому в программе предусмотрены два типа данных о враче, но за отсутствием на данном этапе различий пользователей они не используются.

В результате выделяются 4 основных класса, соответственно для них в главной папке базы должны быть созданы 4 подпапки. Однако для дальнейшего упрощения работы с системой логично добавить ещё 1 класс и соответствующую папку. Данный класс будет содержать данные о всех взаимодействиях врачей с пациентами через таблицу с id кодами для каждой группы (id\_doctor; id\_patient; id\_measurement;). Данная таблица позволит упростить поиск и сортировку данных.

1. Реализация

Рассмотрим по подробней взаимодействие блоков и их заполнение. На первом шаге программа должна произвести поиск папки БД по начальному адресу (адрес программы). Если папка найдена, то перейти ко 2 пункту, в противном случае предлагать пользователю указать адрес расположения папки. Если пользователь укажет директорию без папки, то создать новою и перейти ко 2 пункту.

Второй пункт - это авторизация врача. В окне авторизация предлагается список всех врачей и возможность добавить нового. Выбор осуществляется двойным кликом врача в таблице выбора, на этом 2 этап завершается. В дальнейшем с появление прав добавится ввод пароля, для подключение разрешенных возможностей пользователя.

Третий пункт - это выбор пациента. Поскольку на данном этапе разработки приложения не реализованы другие функции, то единственным возможным выбором пользователя будет выбор пациента. Окно выбора пациента почти ничем не отличается от авторизации, за исключение тем что у пациентов указывается дата рождения. Теперь программа разрешает открыть окно работы с прибором (данный функционал ещё тестируется).

По мимо этапов действий пользователя рассмотрим по подробней что происходит в самой программе. В классе control\_forms происходит работа с основными формами и реализуется их допустимая очерёдность. Например на первом этапе происходит сравнение всех каталогов по стартовому названию с искомым адресом папки БД (!Directory.GetDirectories(address.Replace(@"\БД", "")).Contains(address)), в случае отсутствия принудительно запускается форма Position.

На втором и третьем этапе запускаются разные формы, однако поскольку принцип действия схож, то рассмотрим их обобщённо. В этих формах есть 4 основных функции:

1. Построение списка данных
2. Вывод данных
3. Добавление данных
4. Выбор данных
5. Результаты тестирования