

**ОЛИМПИАДА ШКОЛЬНИКОВ «ШАГ В БУДУЩЕЕ»
НАУЧНО-ОБРАЗОВАТЕЛЬНОЕ СОРЕВНОВАНИЕ
«ШАГ В БУДУЩЕЕ, МОСКВА»**

1086

Регистрационный номер

Информатика и вычислительная техника

название факультета

Системы обработки информации и управления

название кафедры

Анализ корректности распределения нагрузки на тело человека

название работы

Автор:

Овчаренко, Даниил, Валерьевич

фамилия, имя, отчество

ГБОУ Школа №1547, 10 “И”

наименование учебного заведения, класс

Научный руководитель:

фамилия, имя, отчество

место работы

звание, должность

подпись научного руководителя

Анализ корректности распределения нагрузки на тело человека

Аннотация

В современном цифровом мире невозможно представить успешную компанию, деятельность которой не зависела бы от сотрудников умственного труда. Такие компании уделяют большое количество внимания физическому здоровью и эмоциональному состоянию своих сотрудников, т.к. инвестиции работодатели в эти области напрямую влияют на продуктивность и результат.

Одной из самых распространенных причин снижения продуктивности являются проблемы с позвоночником, дисфункции которого сказываются не только на значимом количестве органов человека, но и оказывают существенное влияние на эмоциональное состояние, что в итоге приводит к снижению эффективности.

В рамках текущей работы разработан прототип решения, представляющий собой мат с тензорезистивными датчиками давления, позволяющий отслеживать равномерность распределения нагрузки тела человека на горизонтальную поверхность, и программное обеспечение для анализа осанки.

Данное решение может быть использовано в личных целях, в учебных и государственных заведениях, а также коммерческими компаниями. Данные, собираемые решением, могут быть использованы в аналитических моделях, позволяющих прогнозировать как потенциальные проблемы со здоровьем у сотрудников, так и использовать их для комплексной оценки эффективности сотрудников.

Содержание

1. Исследовательская часть	4
1.1. Актуальность темы и анализ литературы.....	4
1.2. Конкурентный анализ	5
2. Конструкторская часть	6
2.1. Принципиальная схема решения	6
2.2. Описание работы решения	8
2.2.1. Алгоритм работы клиентского контроллера.....	8
2.2.2. Алгоритм работы главного контроллера.....	10
2.2.3. Алгоритм работы управляющего ПО	10
2.2.4. Структура базы данных	11
2.2.5. Отображение данных и отчетность	13
3. Инженерная часть	14
3.1.1. Датчики	14
3.1.2. Клиентский контроллер.....	14
3.1.3. Главный контроллер	15
3.1.4. Радиомодуль	15
3.2. Программные компоненты.....	16
3.2.1. Интегрированная среда разработки	16
3.2.2. Управляющее ПО	16
3.2.3. База данных.....	17
4. Заключение	17
5. Источники	18

1. Исследовательская часть

1.1. Актуальность темы и анализ литературы

Одной из самых распространенных причин снижения продуктивности являются проблемы с позвоночником, дисфункции которого сказываются не только на значимом количестве органов человека, но и оказывают существенное влияние на эмоциональное состояние, что в итоге приводит к снижению эффективности.

По данным ВОЗ, примерно 1,71 миллиарда человек в мире страдают от нарушений и болезней костно-мышечной системы, при этом на люмбаго приходится основная доля общего бремени нарушений и болезней костно-мышечной системы.

В последние годы в литературе широко обсуждалось оптимальное рабочее положение сидя. Согласно исследованиям почти у 18% населения развиваются боли в нижней части спины в течение жизни.

Причиной болей в пояснице является сложный набор анатомических факторов в дополнение к психосоциальным факторам и аспектам рабочей среды. В научных статьях существуют предположения что сочетание неудобного сидячего положения и/или вибрации тела, которая может возникнуть во время вождения на дальние расстояния, с длительным статическим сидячим поведением увеличивает вероятность развития болей. Вполне возможно, что дискомфорт или низкий уровень комфорта, вызванные неблагоприятным или не эргономичным положением сидя, сидячим поведением или условиями работы, могут привести к жалобам со стороны опорно-двигательного аппарата.

При нарушениях осанки деформируется скелет, нагрузка на суставы, связки, мышцы распределяется неправильно, из-за чего страдает весь опорно-двигательный аппарат, ухудшается рессорная функция позвоночника. Снижение функции пружины позвоночника приводит к стойким микротравмам головного и спинного мозга при ходьбе, беге и других движениях, что негативно влияет на высшую нервную деятельность, сопровождающейся снижением работоспособности. Кроме того, при наличии дефектов осанки внутренние

органы могут отклоняться от нормы положений и быть зажатыми другими органами и тканями. Спинальный мозг, расположенный в позвоночнике, участвует в большинстве рефлексов. Если положение позвоночника нарушено, спинномозговые нервы зажаты, кровообращение нарушается ликвор, что негативно сказывается на всем организме.

Боль в спине обычно развивается в подростковом возрасте и к позднему подростковому возрасту достигает уровня распространенности, сходного с таковым у взрослых. Более глубокое понимание поведенческих механизмов ответственных за развитие болей в спине необходимо человеку в том числе для информирования лечащего врача, который может назначить методы самостоятельной коррекции пациента в положении сидя.

Наиболее рациональным подходом в борьбе с проблемами спины является превентивный подход, в рамках которого можно получить данные ещё до наступления ухудшений, либо на начальных стадиях и вовремя предпринять меры.

В рамках текущей работы разработан прототип решения, представляющий собой мат с тензорезистивными датчиками давления, позволяющий отслеживать равномерность распределения нагрузки тела человека на горизонтальную поверхность, и программное обеспечение для анализа осанки.

Данное решение может быть использовано в личных целях, в учебных и государственных заведениях, а также коммерческими компаниями. Данные, собираемые решением, могут быть использованы в аналитических моделях, позволяющих прогнозировать как потенциальные проблемы со здоровьем у сотрудников, так и использовать их для комплексной оценки эффективности сотрудников.

1.2. Конкурентный анализ

В ходе выполнения работы был проведён поиск аналогов и произведен сравнительный анализ решений. Результаты анализа приведены в таблице.

Разработанное в ходе работы решение обладает преимуществами перед конкурентами и может быть коммерчески успешно.

Функция	Проект	Мастер осанки	Darma	UpRight
Использование базы данных	+	-	-	-
Просмотр загруженных и ненагруженных областей	+	-	+	-
Составление отчетов по показаниям	+	-	+	+
Возможность ношения на туловище	-	+	-	+
Приложения для компьютера	+	-	-	-
Возможность подключения нескольких клиентов	+	-	-	-

2. Конструкторская часть

2.1. Принципиальная схема решения

Для реализации решения была выбрана клиент-серверная архитектура, т.к. она позволяет горизонтально масштабировать количество клиентов без изменения логического серверного слоя.

Решение состоит из следующих логических блоков:

- Клиент
- Главный контроллер
- Сервер приложений
- Пользовательское ПО

Каждый клиент состоит из 10 тензорезисторов, подключенных к контроллеру, оснащеному радиомодулем. Контроллер с заданной периодичностью считывает показания датчиков, с последующим сохранением полученных значений в оперативную память и передачей их по радиоканалу на главный контроллер.

Главный контроллер оснащен радиомодулем и имеет физическое подключение к аппаратной части сервера приложений по интерфейсу USB. Главный контроллер хранит в оперативной памяти информацию, поступающую с клиентских контроллеров.

На сервере приложений размещено управляющее ПО и база данных. Управляющее ПО с заданной периодичностью инициирует получение данных с главного контроллера и осуществляет их сохранение в базу данных.

Пользовательское ПО предназначено для чтения и визуализации данных из базы данных.

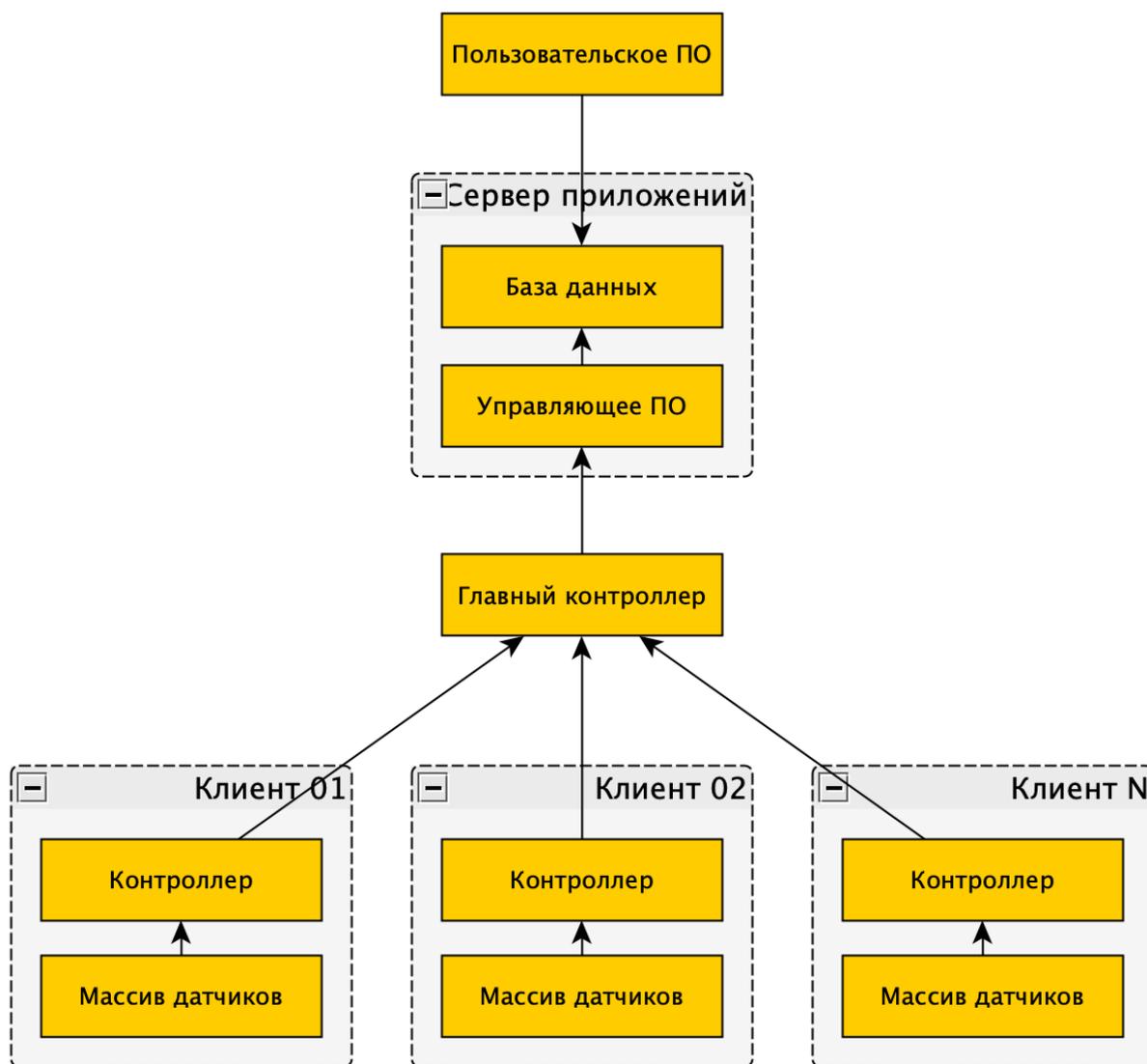


Рисунок 1 – Принципиальная схема решения

2.2. Описание работы решения

2.2.1. Алгоритм работы клиентского контроллера

Разработанный алгоритм записан в постоянную память клиентского контроллера и выполняется при наличии питания клиентского контроллера.

- Инициализация портов, к которым подключены датчики
- Получение значений с датчиков, преобразование и запись данных в массив, хранящийся в ОЗУ клиентского контроллера
- Передача массива данных по радиоканалу на главный контроллер

Листинг инициализации портов

```
uint8_t ports[portsSize] = {A0, A1, A2, A3, A4, A5, A6, A7, A8, A9};

for (int counter = 0; counter < portsSize; counter++) {
  data[counter] = 0;
  pinMode(ports[counter], INPUT);
}
```

Получаемые контроллером данные с датчиков подлежат преобразованию к процентным значениям для удобства последующей обработки.

Листинг функции получения данных и преобразования

```
void startAnalyze() {
  float path = 0;
  for (byte counter = 0; counter < portsSize; counter++) {
    path = (analogRead(ports[counter])); //Считываем во временную переменную значения с порта
    path /= 1023.0; //Делим на разрядность порта Arduino
    path = 1 - path; //Для определения степени нажатия, вычитаем из полной загрузки текущее значение
    path /= 4; //Сопротивление тензорезистора не способно полностью уменьшить напряжения, делим на 4
    path *= 100; //Переводим в проценты
    data[counter] = (byte) path; //Записываем в массив.
  }
  delay(1000);
}
```

Для реализации работы с радиомодулем была использована библиотека VirtualWire.

Листинг функции передачи данных через радиоканал

```
void startRadio() {
  digitalWrite(13, HIGH); //Включаем светодиод, на микроконтроллере
  byte toSend[dataSize + 1]; //Создаем массив с данными и ID
  toSend[0] = 0; //Присваиваем идентификационный номер
  for (byte counter = 1; counter < dataSize + 1; counter++) {
    toSend[counter] = data[counter - 1]; //Заполняем массив данными
  }
  vw_send(toSend, 11); //Отправляем массив
  vw_wait_tx(); //Ждем передачи
  digitalWrite(13, LOW); //Гасим светодиод
}
```

2.2.2. Алгоритм работы главного контроллера

Главный контроллер выполняет роль связующего звена между клиентскими контроллерами и управляющем ПО. На нём реализованы следующие функции:

- Сохранение данных с клиентских контроллеров при получении радиосигнала
- Передача данных в управляющее ПО при получении запроса от последнего

Листинг функции получения данных с клиентских контроллеров

```
void startListeningRadio() {
    vw_rx_start(); //Ждем получения сообщения
    buflen = VW_MAX_MESSAGE_LEN; //Задаем переменную, содержащую максимальную длину
    сообщения
    if (vw_have_message()) { //Если видим сообщения, ждем пока оно полностью будет передано
        if (vw_get_message(buf, & buflen)) { //Записываем сообщения в массив - буфер
            vw_rx_stop(); //Останавливаем прием
            for (byte counter = 1; counter < 12; counter++) {
                data[buf[0]][counter - 1] = buf[counter]; //Сохраняем данные в массив
            }
        }
    }
}
```

Листинг функции отправки данных в управляющее ПО

```
void startListener(char message) {
    if (message == 'g') { //Если получили сообщение от главного клиента
        for (byte clientCounter = 0; clientCounter < 10; clientCounter++) { //Выводим построчно двумерный
        массив
            for (byte counter = 0; counter < 10; counter++) {
                Serial.print(data[clientCounter][counter]);
                Serial.print(";");
            }
            Serial.println();
        }
        Serial.println();
    } else if (message == 'c') { //Команда проверки подключения
        Serial.println("Connected!");
    }
}
```

2.2.3. Алгоритм работы управляющего ПО

Управляющее ПО выполняет функции:

- Получения данных с главного контроллера
- Сохранение данных в базу данных
- Построение отчетов

Класс `ArduinoController`, обеспечивает возможность работы с главным контроллером.

Класс `ArduinoConnection` обеспечивает возможность получения данных.

Метод `checkConnection` обеспечивает проверку доступности главного контроллера:

- Приложение с помощью библиотеки `Arduino` обращается к главному клиенту, посылая сообщение 'g'
- Если ответ не был получен, ждет 500 миллисекунд и отправляет запрос снова, если после 6 попыток не был получен ответ, программа возвращает ошибку

Метод `carryInfoFromClient` обеспечивает получение данных с главного контроллера и создание объектов на основе POJO класса `Report`

Метод `insertReportToDb` обеспечивает запись объектов `Report` в базу данных

2.2.4. Структура базы данных

Структура базы данных состоит из следующих таблиц:

`chair_owner` – Таблица с информацией о клиенте и владельце

Атрибут	Тип	Описание
<code>chair_id</code>	INTEGER	ID кресла
<code>owner_id</code>	INTEGER	ID Владельца

`chair_info` – Таблица показаний датчиков

Атрибут	Тип	Описание
chair_id	INTEGER	ID кресла
date_of_indication	TIMESTAMP_WITH_TIMEZONE	Время снятия показаний
snumber_sensor	INTEGER	Показания датчиков

owner_info – Таблица с информацией о владельце

Атрибут	Тип	Описание
owner_id	INTEGER	ID Владельца
owner_name	TEXT	Имя владельца
owner_surname	TEXT	Фамилия владельца

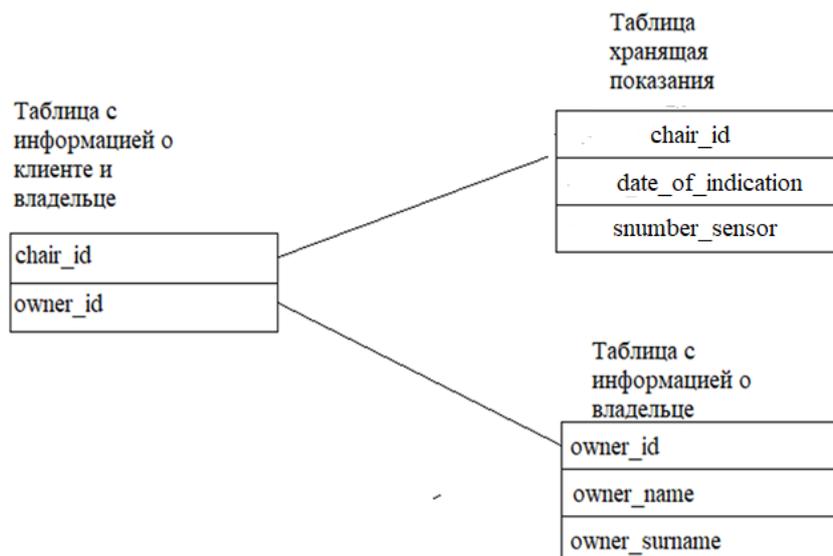


Рисунок 2 – Схема базы данных

2.2.5. Отображение данных и отчетность

В разработанном ПО доступен просмотр графиков отклонений загрузки датчиков по линиям соприкосновения с телом человека. По оси X отражено время измерения, по оси Y – нагрузка. Большая нагрузка на правую сторону отображается в положительной части оси Y- над осью X нагрузка на левую сторону - под осью X.



Рисунок 3 – Графики загруженности датчиков

3. Инженерная часть

3.1. Аппаратные компоненты

3.1.1. Датчики

Основная задача датчиков – снятие показаний текущей нагрузки на ограниченную область. Для решения этой задачи выбор был остановлен на тензорезисторных датчиках.

На момент выполнения научной работы в свободной продаже был доступен ограниченный ряд моделей. После детального изучения список был сокращён до двух моделей, максимально подходящих по своим параметрам для решения поставленной задачи:

- RFP602
- FSR402

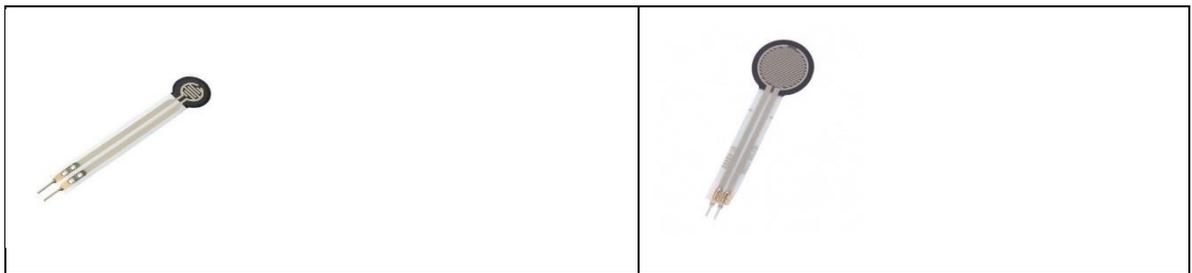


Рисунок 4 – Внешний вид датчиков

3.1.2. Клиентский контроллер

Выбор решения для клиентского контроллера производился на основании следующих критериев:

- Возможность подключения 10 и более датчиков
- Наличие возможности разработки на языках C/C++
- Малое энергопотребление
- Компактность
- Доступность на рынке
- Простота в освоении
- Широкий выбор готовых библиотек
- Наличие базы знаний и активного сообщества разработчиков

Наиболее подходящим решением, удовлетворяющим требованиям, является контроллер Arduino Mega 2560 Pro Embed (Рис. 4), который и был использован в проекте.

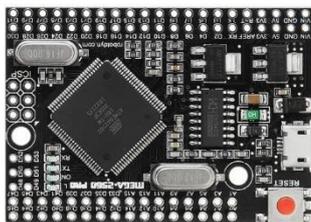


Рисунок 5 - Arduino Mega 2560 Pro Embed

3.1.3. Главный контроллер

При выборе главного контроллера были использованы аналогичные требования, применяемые к клиентскому контроллеру, а также дополнительные требования:

- Подключение периферийных модулей без использования пайки
- Наличие интерфейса USB

В целях снижения рисков совместимости был выбран контроллер того же производителя, что и клиентский контроллер - Arduino Uno R3



Рисунок 6 - Arduino Uno R3

3.1.4. Радиомодуль

Выбор радиомодуля производился на основании следующих критериев:

- Совместимость с клиентским и главным контроллерами
- Рабочая частота 433 МГц
- Доступность на массовом рынке

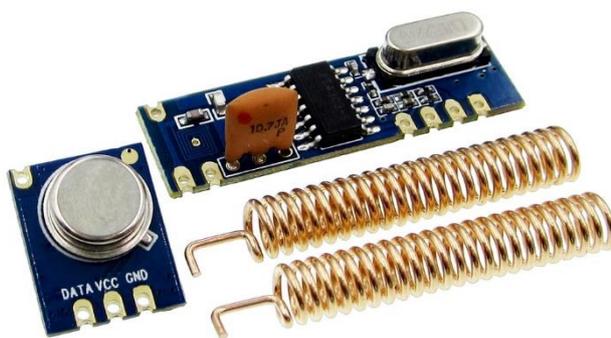


Рисунок 7 – Радиомодуль. Приёмник и передатчик.

3.2. Программные компоненты

3.2.1. Интегрированная среда разработки

Основными языками разработки на платформе Arduino являются C и C++.

Был проведен сравнительный анализ трёх наиболее распространённых интегрированных средств разработки:

- Visual Studio Code
- Clion
- Arduino IDE

Выбор был сделан в пользу Visual Studio Code и дополнения PlatformIO, т.к. данные продукты являются бесплатными, имеют активные сообщества разработчиков и поддержку платформы на необходимом для проекта уровне.

3.2.2. Управляющее ПО

При выборе языка программирования для реализации управляющего были использованы следующие критерии:

- Наличие бесплатных библиотек для работы с платформой Arduino
- Кросс-платформенность
- Возможность использования языка для дальнейшей реализации веб-клиента
- Степень владения выбранным языком

В результате анализа и сопоставления критериев, для разработки управляющего ПО был выбран язык программирования Java, и библиотеки:

- Java-Arduino Communication Library

- Javafx

В качестве интегрированной среды разработки была выбрана IntelliJ IDEA и решение SceneBuilder для ускорения реализации экранных интерфейсов.

3.2.3. База данных

Для хранения данных была выбрана реляционная бесплатная база данных PostgreSQL.

4. Заключение

В ходе проведенной работы достигнута поставленная цель и создан прототип решения, позволяющий определить корректность осанки человека, а также собирать данные о длительности пребывания человека в сидячем положении. Решение обладает низкой себестоимостью и богатыми возможностями по масштабированию как в программной части, так и в областях применения. К таким возможностям относятся:

- Создание мобильного приложения для личного пользования
- Интеграция решения в существующие приложения класса Health Care
- Коллаборации с производителями офисной мебели
- Использование данных в комплексной аналитике эффективности сотрудников
- Использование учебными заведениями при дистанционном обучении
- Использование в общественных местах
- Использование в ресторанном бизнесе
- Технология может быть применена в тренировках различных видов спорта, успешность в которых связана с корректным распределением веса, таких как сноуборд, скейтборд, серфинг, занятия йогой.

5. Источники

1. П. Винк, С. Холлбек От редакции: исследования комфорта и дискомфорта демонстрируют потребность в новой модели. *Аплл. Эргон.*, 43 (2012).
2. А.М. Лис, К.М. Блэк, Х. Корн, М. Нордин Связь между сидячей и профессиональной БНС Евро. Спайн , 2007.
3. Лайонел К.А. Факторы риска хронической боли в пояснице у взрослых: исследование «случай-контроль», проведенное в Шри-Ланке. J Облегчение боли. 2014.
4. Питер Б. О'Салливан, Энн Дж. Смит, Даррен Дж. Билз, Леон М. Стрейкер Ассоциация биопсихосоциальных факторов со степенью сутулости в сидячей позе и самооценкой боли в спине у подростков: кросс-секционное исследование, *Физиотерапия*, том 91, выпуск 4, 2011 г.
5. Тюрин Т., Мазенников С. Исследовательская работа по теме «Изучение проблем возникающих при нарушении осанки и методов ее профилактики»
6. Кабышева, М. И. Функциональные нарушения опорно-двигательного аппарата студентов / М.И. Кабышева, Т.А. Глазина // *Вестник Оренбургского университета*, 2017. – № 2 (202). –
7. И.В. Рубцова, Т.В. Кубышкина, Н.В. Лукшина Осанка. Средства и методы оценки и коррекции. Учебно-методическое пособие для вузов, 2008.