Разработка программы для обработки физиологических данных о сне человека с целью определения оптимального момента его пробуждения.

Соискатель: студент группы ИВБ1-19 Чунаков М. А.

Руководитель: ст. преп. Каленов А. С.

Описание предметной области



Студентам важно запоминать информацию, которую дают на парах. Если человек чувствует себя сонным, то его внимание рассеяно. Когда внимание человека рассеянно он хуже запоминает информацию. Следовательно, если студент не высыпается то он хуже запоминает информацию, поэтому данная программа будет полезна студентам.

Работники работающие в основном с компьютерами

Работникам работающих с компьютерами важно быть внимательными. Если человек будет чувствовать себя сонным, то он будет не внимателен и может допустить ошибки в заполнении каких-либо данных или при написании программы. Вследствие чего данная программа будет полезна людям работающим на компьютере.

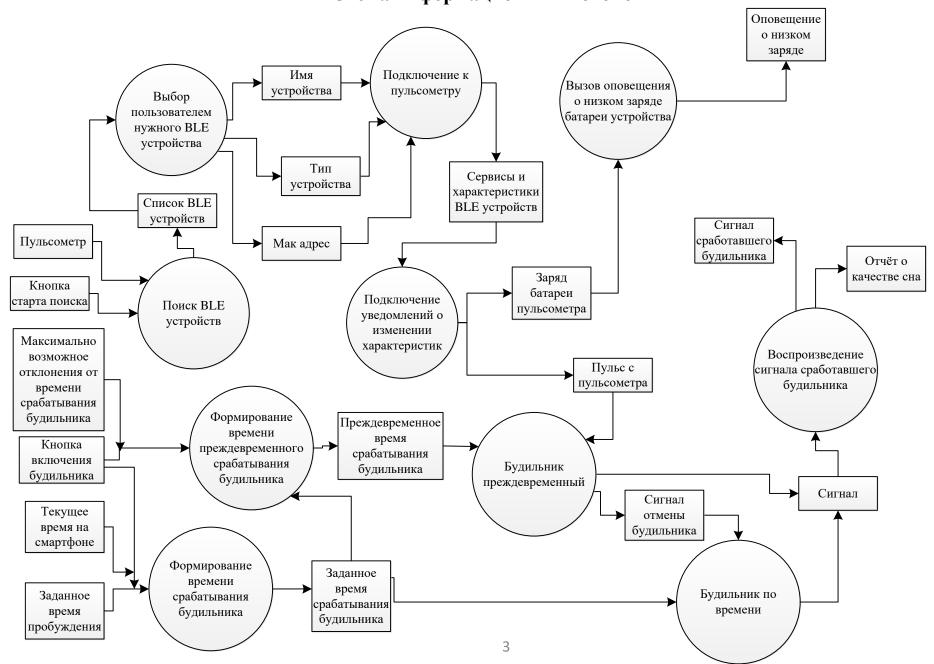


Работники работ с уклоном в физический труд



Работникам работающих на работах с уклоном на физический труд важно оставаться дольше не уставшим. Если человек чувствует себя сонным, то он быстрее устаёт. Вследствие чего человеку требуется больше времени, для того чтобы отдохнуть после работы. Поэтому данная программа будет полезна людям работающим на работах с уклоном на физический труд

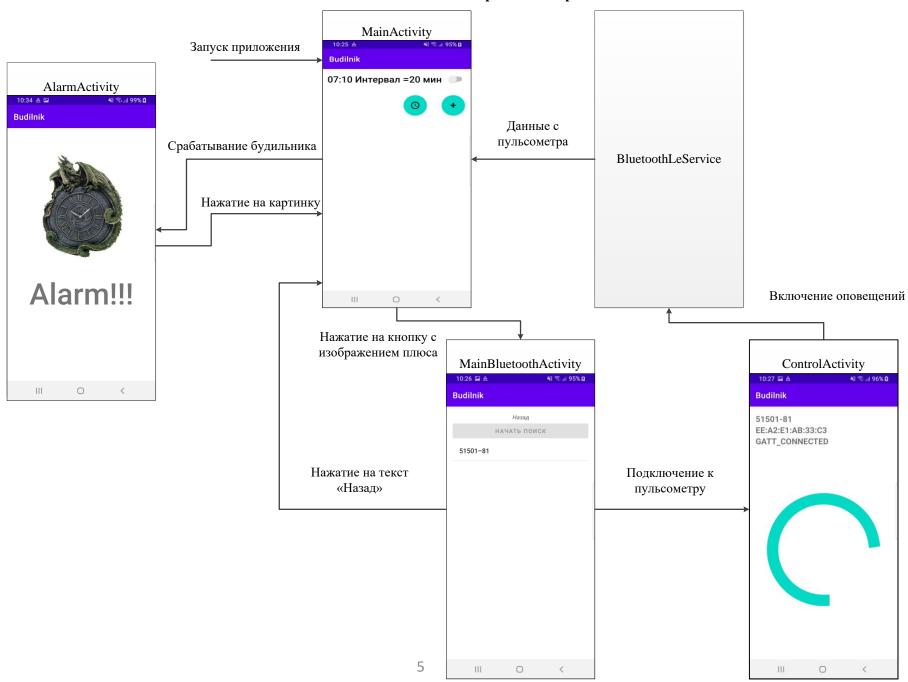
Схема информационных потоков



Сравнения аналогов

Название варианта	Изображение устройства	Плюсы	Минусы
Fitbit		Удобное использование	Дороговизна
		Имеются другие функции	Отдельное устройство
			Не определяет фазы сна
Misfit Shine 2		Удобное использование	Дороговизна
		Имеются другие функции	Отдельное устройство
		Определяет фазы сна	
Приложения для отслежывание качества сна	Standard Commence of the Control of	Собирает и анализирует	Не выполняет функцию
		информацию о качестве сна	будильника
		Существует на все системы смартфонов (<i>Android</i> и <i>IOS</i>)	Не все функции доступны бесплатно
Приложение Xsmart		Определяет фазы сна	Сложная настройка
		Множество дополнительных функций	Работает только с браслетом Мі Вапд 2
		функции	Dana 2
			Существует только для Android

Блок схема взаимодействия процессов приложения



Статическая модель программы

Нажатие на имя устройства и в открывшемся Нажатие на кнопку с + Подключение оповещений об изменении значений характеристик устройсва диалоговом окне на «Подключиться» Класс MainBluetooth Класс ControlActivity Класс BluetoothLeService Отвечает за поиск BLE устройств Отвечает за подключение к устройству Отвечает за обработку поступающих с пульсометра данных. Инициализация переменных: Инициализация переменных: Инициализация переменных: listBluetoothDevice;//хранит mBatteryNotifyCharacteristic;//Переменная хранящая batteryPct=0; //Переменная хранящая заряд смартфона data: найденные устройства данные для оповещения о изменении //Массив для получения входящих данных Класс MainActivity listBluetoothDeviceString://хранит характеристики(Заряда батареи устройства) // Название каналов уведомлений Отвечает за установку и включение имена найденных устройств mPulsNotifyCharacteristic;//Переменная хранящая CHANNEL ID = "Оповещения о входящих данных"; будильника и расчёт минимального adapterLeScanResult;//Промежуточная данные для оповещения о изменении CHANNELZU_ID = "Оповещение о малом уровне заряда пульсометра"; времени пробуждения CHANNELZT ID = "Оповещение о недостаточном уровне заряда телефона"; переменная нужная для передачи характеристики (Пульс пользователя) Инициализация переменных: данных от listBluetoothDeviceString в mGattServicesList;//Хранит перечень сервисов и CHANNELO_ID = "Оповещение об отключении пульсометра от телефона"; heartRate;//Переменная хранящая listViewLE характеристик устройства heartRate: //Хранит полученное значение пульса пульс пользователя mHandler;//переменная обработчик mConnectionState = STATE DISCONNECTED;//Хранит текущую стадию подключения sdf = new SimpleDateFormat("dd LLLL процесс поиска устройств Метолы класса: timer=0: //переменная счётчик используется для задержки между HH:mm:ss", Locale.getDefault()):// SCAN PERIOD = 20000;//сколько updateConnectionState(final String st)//Метод оповещениями о входных данных Переменная хранящая вормат даты и максимально будет длиться поиск выводящий стадию подключения на экран //UUID Сервисов и характеристик времени Методы класса: onCreate(@Nullable Bundle savedInstanceState)// //Заряд батареи //Переменные хранящие заданное Battery Level UUID = UUID.fromString("00002a19-0000-1000-8000-00805f9b34fb"); onCreate(Bundle savedInstanceState)// Метод вызывающийся при переключении на время будильника данный класс //Имя устройства Метод вызывающийся при public static Integer hour://Часы onDestrov()//Метод вызывающийся при переключении на данный класс; String Name ledService = "00001800-0000-1000-8000-00805f9b34fb": public static Integer min://Минуты String Name switchChar = "00002a00-0000-1000-8000-00805f9b34fb"; OnItemClickListener()//Метод переключении с этого класса на другой public static Integer day://День oBacking(View view)//Метол вызывающийся при //Пульс отвечающий за то что произойдёт //Переменные хранящие вычисленное нажатии на кнопку «на главный экран» UUID Name switchChare =UUID.fromString(String Name switchChar); когда нажмут на имя устройства в минимальное время срабатывания списке найденных устройств BackingGl()//Метод возвращающий на главный String Puls ledService ="0000180d-0000-1000-8000-00805f9b34fb"; будильника scanLeDevice(final boolean enable)// экран ParcelUuid Puls ledService =ParcelUuid.fromString(String Puls ledService): public static Integer hour2;//Часы String Puls switchChar = "00002a37-0000-1000-8000-00805f9b34fb"; Метод производящий поиск public static Integer min2;//Минуты vстройств UUID Puls switchChare =UUID.fromString(String Puls switchChar); public static Integer day2;//День oBacking(View view)//Метод zaryd=0; //переменная хранящая заряд устройства priv=0; отвечающий за то что произойдёт //счётчик использующийся для подтверждения быстрой фазы сна Методы класса: когда нажмут на текст «Назад» Метолы класса: onCreate(Bundle savedInstanceState)// broadcastUpdate(final String action, final BluetoothGattCharacteristic characteristic)//Метод Метод вызывающийся при вызывающийся при изменении характеристики переключении на данный класс Преждевременное createChannelIfNeeded(NotificationManager manager)//Метод создающий каналы для fab.onClick(View view)//Метол срабатывание будильника оповещений отвечающий за то что произойдёт при setCharacteristicNotification(BluetoothGattCharacteristic characteristic, boolean enabled)// нажатии на кнопку с плюсом Нажатие на текст «Назад» Включения оповещения об изменении характеристики Interval2.onClick(View view)//Метол notifyk()//Оповещения о входящих данных отвечающий за то что произойдёт при Нажатие на кнопку «На главный экран» Класс AlarmActivity notifykZP()//Оповещение о малом уровне заряда пульсометра нажатии на кнопку с часами или завершение подключения -класс отвечающий за индикацию того что notifvkZT()//Оповещение о нелостаточном уровне заряда телефона switchView.onCheckedChanged(Compo оповещений об изменении значений сработал будильник. notifykO()//Оповещение об отключении пульсометра от телефона undButton buttonView, boolean характеристик устройства Инициация переменных класса: budilnik()//Расчёт преждевременного срабатывания будильника isChecked)//Метод отвечающий за то // переменная в которой хранятся часы hour: что произойдёт когда переключится Срабатывание будильника установленного будильника; переключатель min; // переменная в которой хранятся Нажатие на картинку budilka()//Включение будильника минуты установленного будильника: otmena()//Отмена включённого Методы класса:

> onCreate() //Метод вызывающийся при переключении на данный класс; cansel() //Метод выключающий рингтон и

otmena() //Метод отменяющий запланированный будильник если класс вызвали

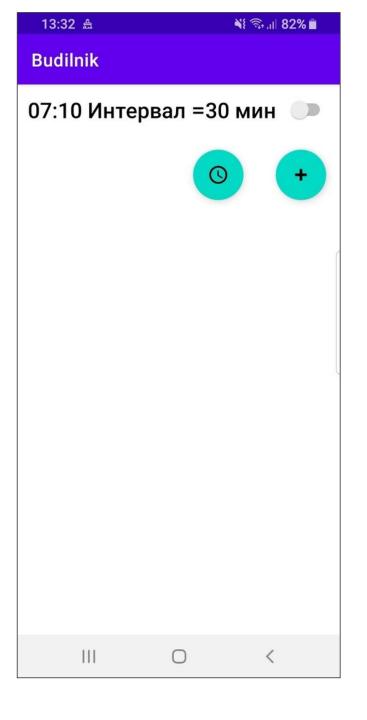
вибрацию; otmena()

досрочно;

будильника

При запуске приложения открывается такой экран.

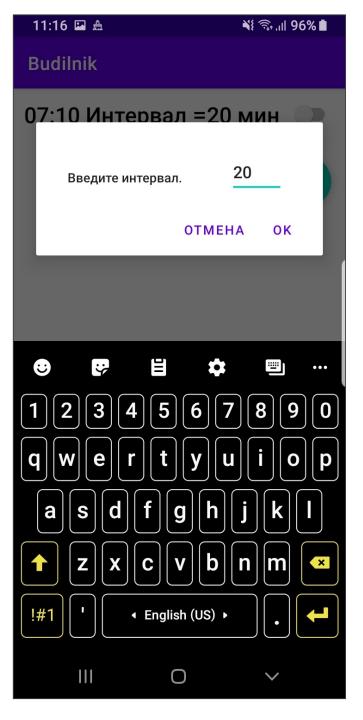
Для изменения интервала между минимальным и максимальным временем срабатывания будильника надо нажать на кнопку с изображением часов.



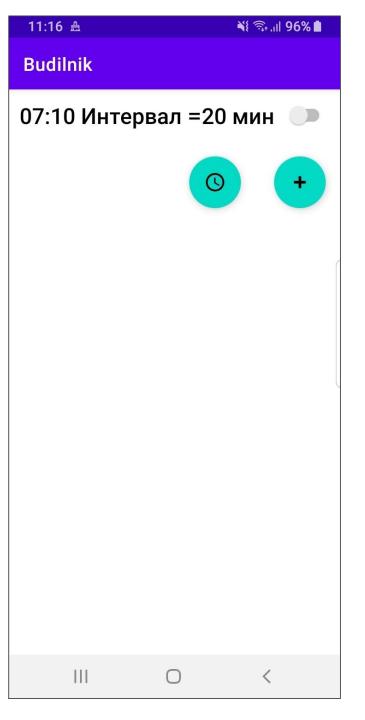
После нажатия кнопки выведется такое диалоговое окно. Для ввода нужно нажать на чёрточку. Выбор «отмены» вернёт вас на начальный экран.



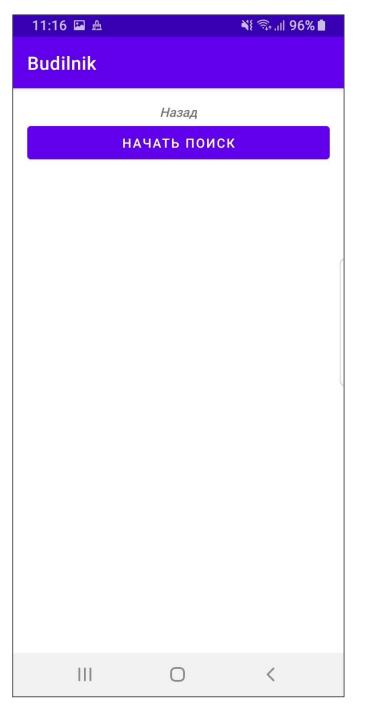
После ввода желаемого интервала (в минутах) и нажатия «ОК» вас вернёт на начальный экран и изменит интервал на выбранный.



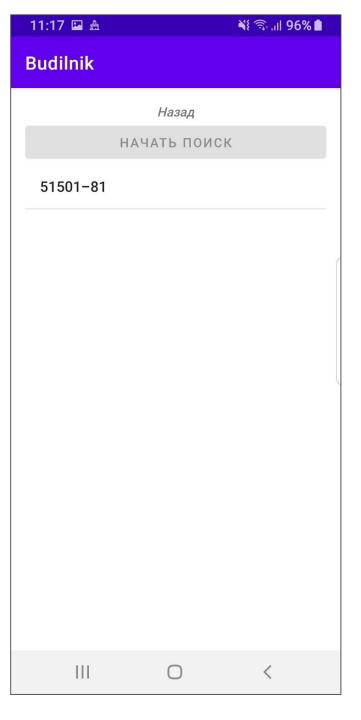
Для подключения пульсометра нужно нажать на кнопку с изображением плюса.



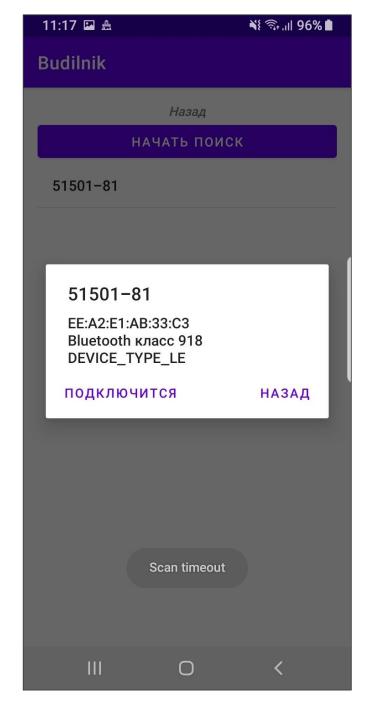
После нажатия на кнопку откроется этот экран. Для старта поиска включённого пульсометра нажмите кнопку «Поиск». При нажатии на текст «Назад» откроется начальный экран.



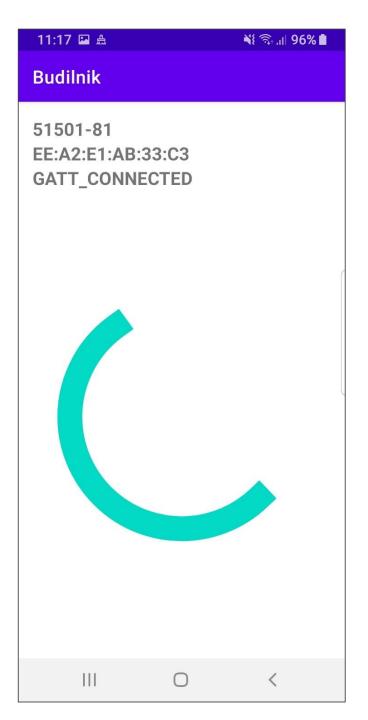
После того как программа найдёт ваш включённый пульсометр выберете его имя в списке найденных устройств.



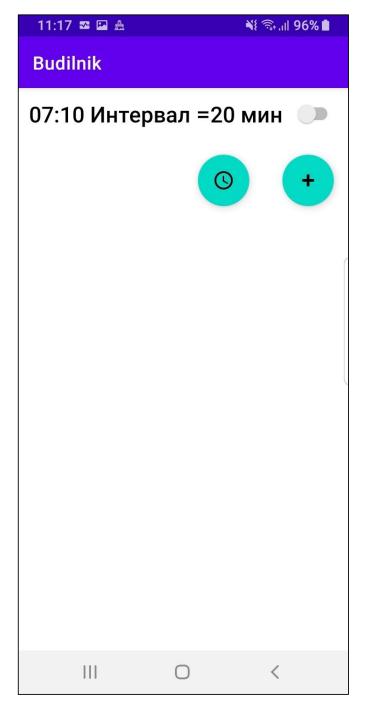
После нажатия на название нужного устройства появится данное диалоговое окно. При нажатии кнопки «Назад» вас вернёт на экран поиска ВLЕ Устройств. При нажатии кнопки «Подключится» программа перейдёт на экран процесса подключения к выбранному устройству.



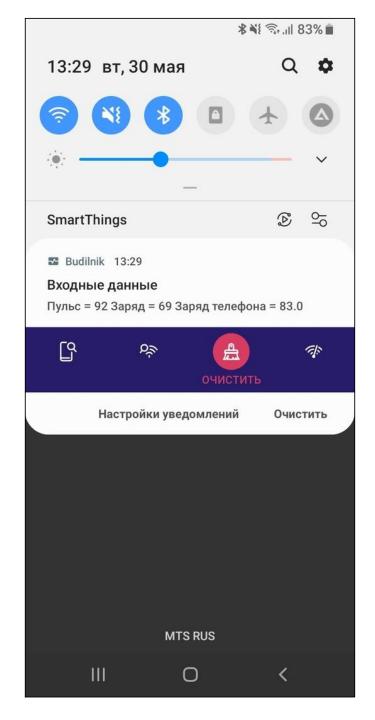
Вот так выглядит экран процесса подключения к устройству. После того как программа подключится к устройству программа вернёт вас на начальный экран, если подключится не получилось, то появится кнопка назад при нажатии на которую вас вернёт к поиску ВLE устройств.



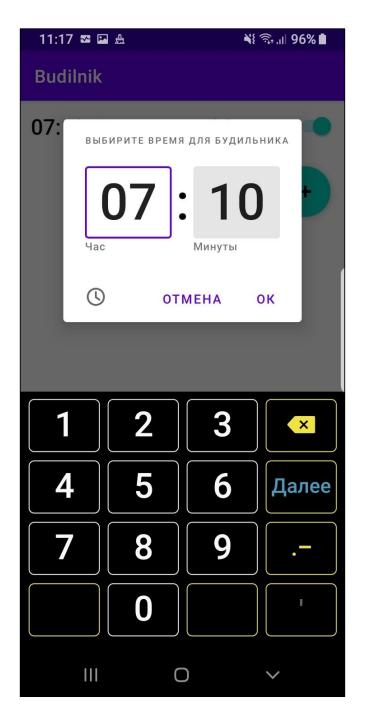
После подключения к устройству в уведомлениях телефона появится новое уведомление. Для включения будильника нужно нажать на переключатель находящийся над кнопками.



Вот так выглядит уведомление.



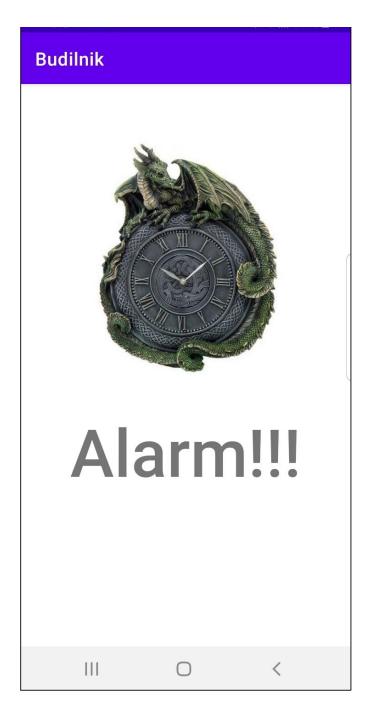
После нажатия на переключатель появится вот такое диалоговое окно для ввода времени срабатывания будильника. Если устанавливаемое время меньше или равно текущему то будильник устанавливается на следующий день.



После нажатия на кнопку «ОК» вас вернёт на начальный экран и напишет на сколько установлен будильник.



После того как сработает будильник откроется этот экран, телефон начнёт вибрировать и если включён начнёт играть музыка рингтона. При нажатии на картинку вас вернёт на начальный экран.



Спасибо за внимание