OUR SOLUTION

DOC HERO

make the world a sweetest place

Team: Mostaf Ramezani,Soma Shekarchi

Problem

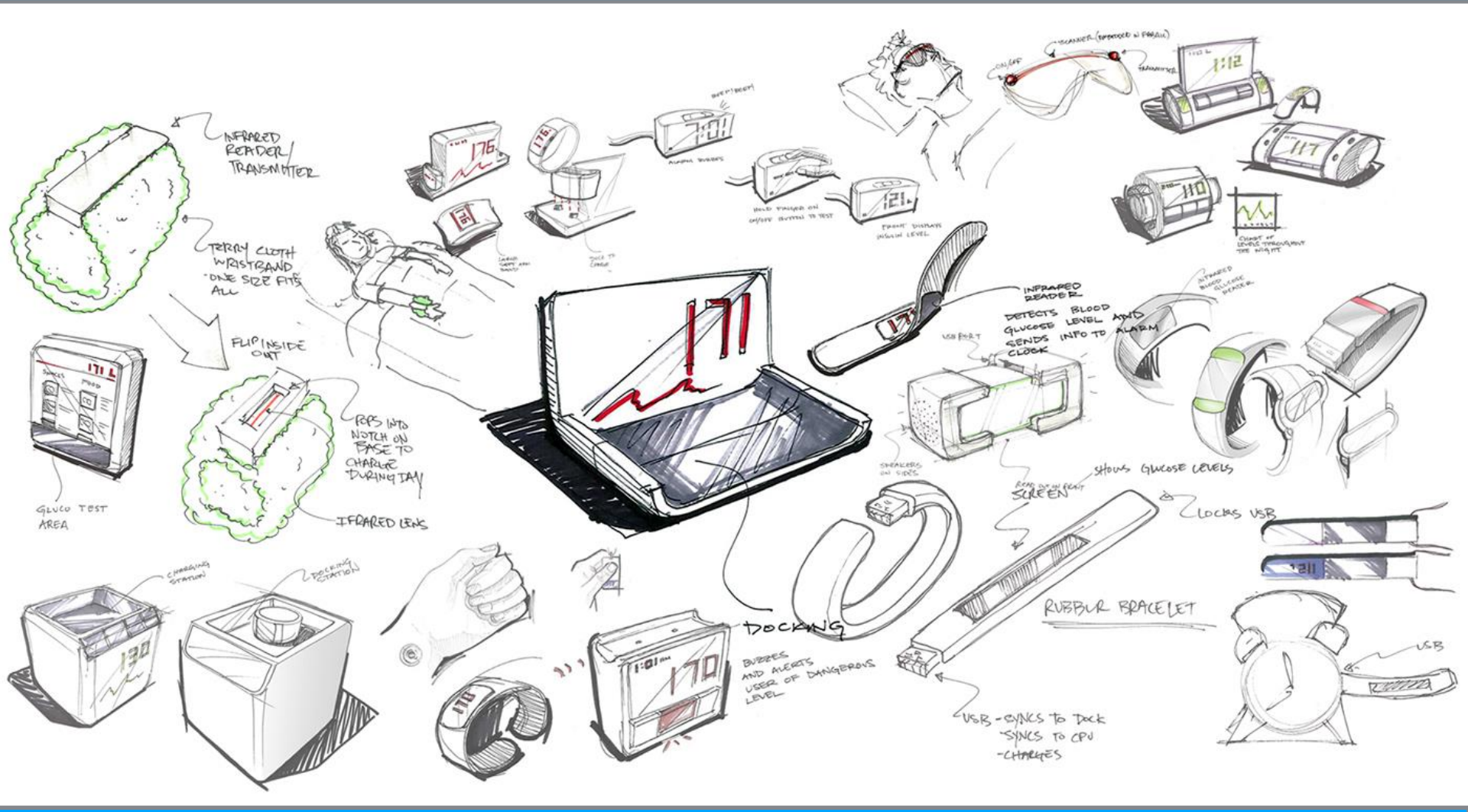
Low blood sugar levels, known as hypoglycemia, in people with diabetes may cause potentially dangerous changes in heart rate. Someone with type 1 diabetes is found dead in the morning in an undisturbed bed after having been observed in apparently good health the day before.

Solution

A wearable measuring the beat-to-beat variation in heart rate is a promising device for the early detection of hypoglycemia, or low blood sugar, in type 1 diabetes.
System sends immediate alerts to your smart device or receiver when your heartrate is trending too high or too low.

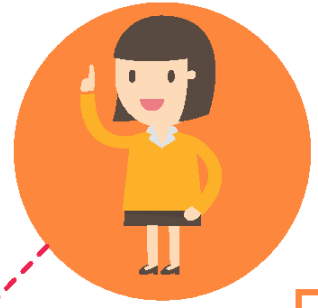
STORYBOARD

DOC HERO
make the world a sweetest place

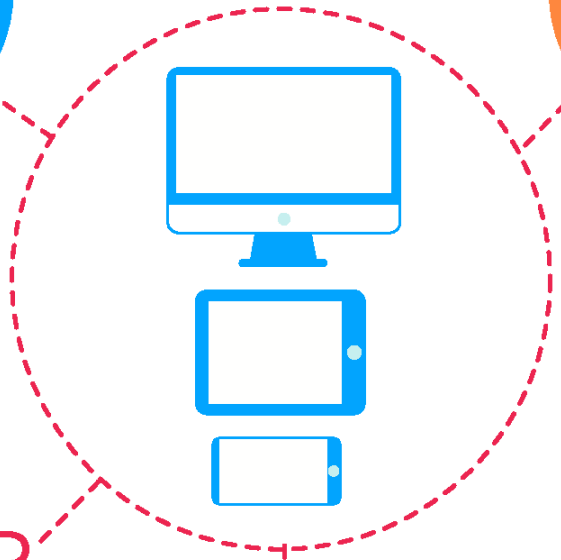


THE ACTORS

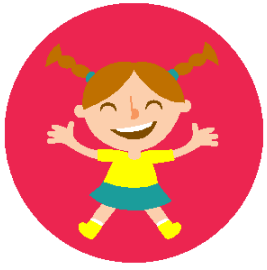
DOCTORS



PARENT



MVP



CHILDREN

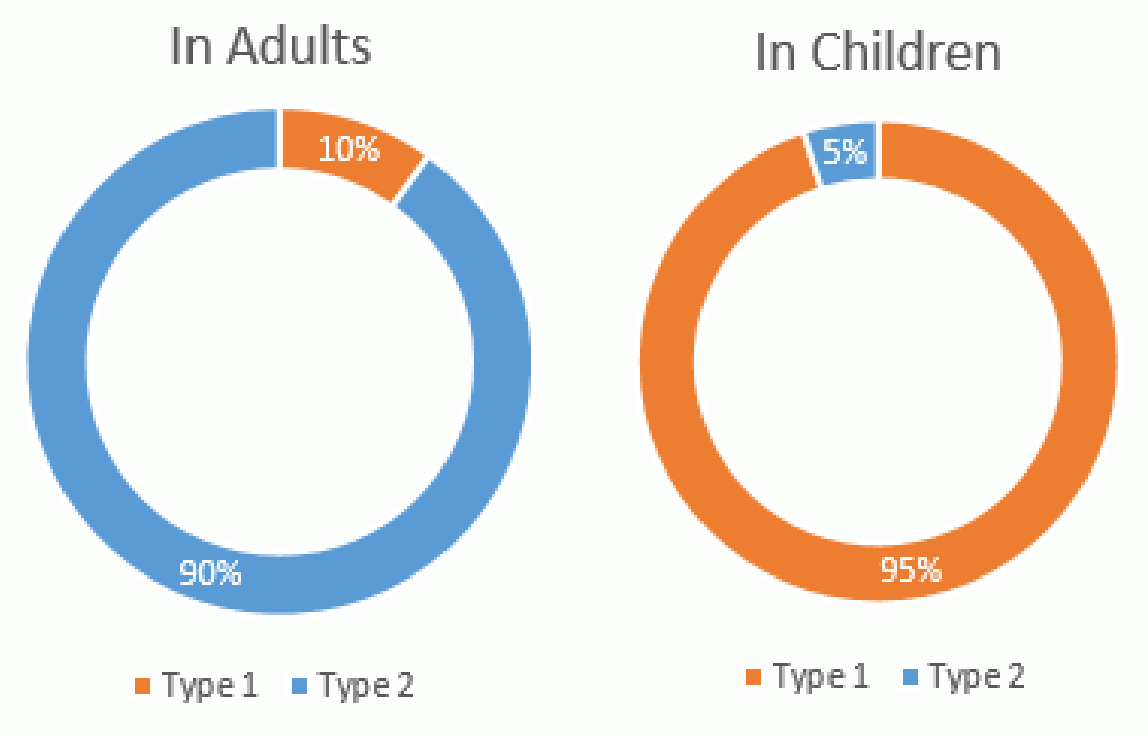
Diabetes Children Statistics

Table 3.9 Global estimates of type 1 diabetes in children (<15 years) for 2015

Child population (< 15 years)	1.9 billion
Type 1 diabetes in children (< 15 years)	
Number of children with type 1 diabetes	542,000
Number of new type 1 diabetes cases per year	86,000
Annual increase in incidence	3%*

* Estimate from the *Diabetes Mondiale study (DIAMOND)*[24], the *Europe and Diabetes study (EURODIAB)*[25].

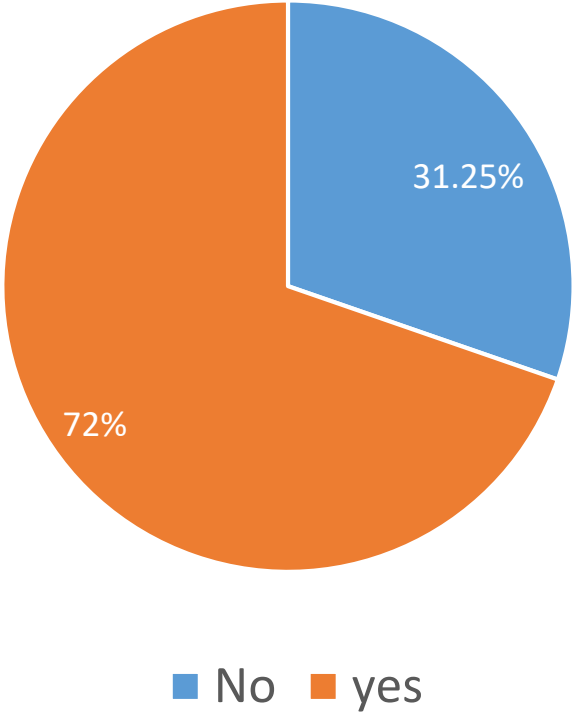
Diabetes Children Statistics



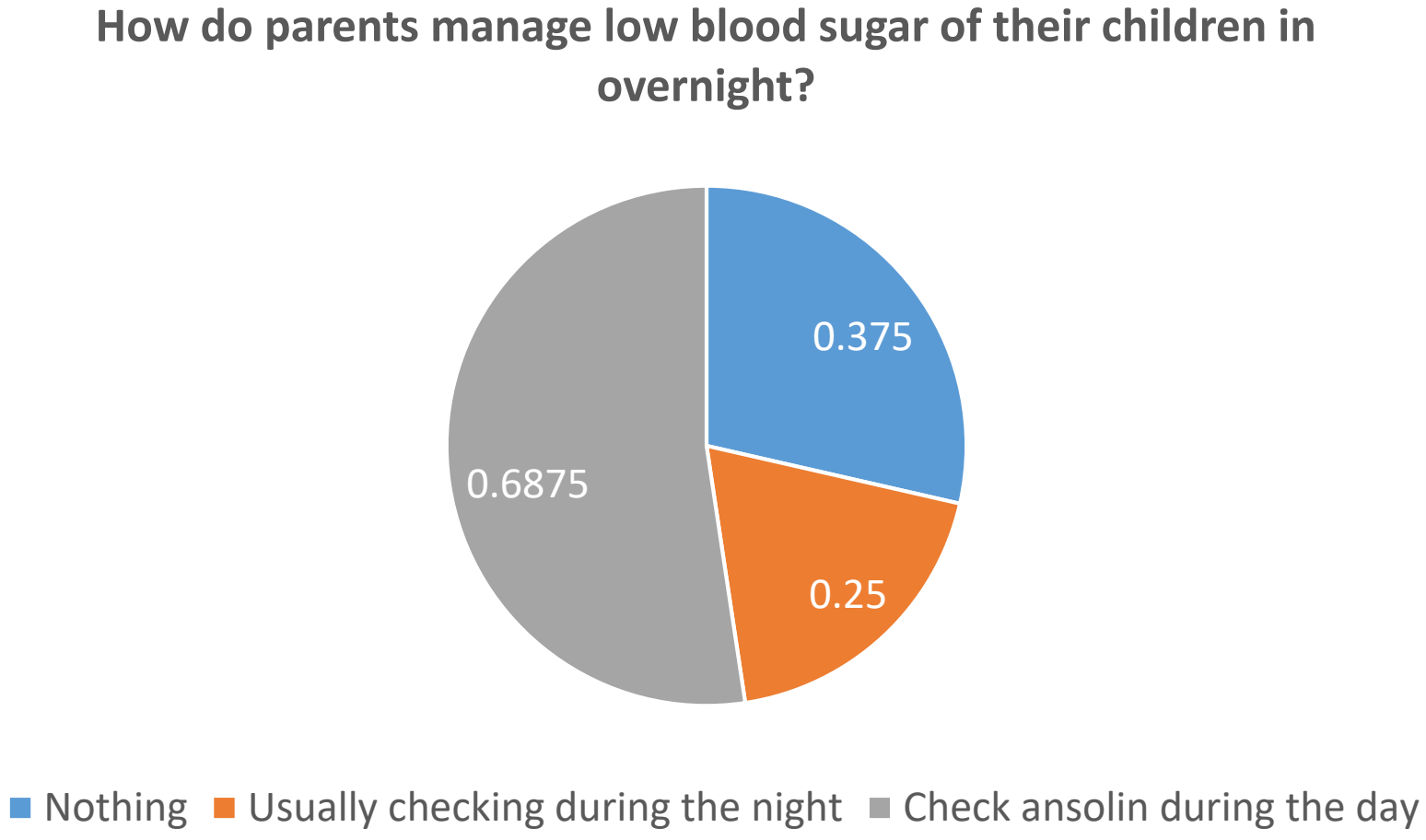
Problem Validation

make the world a sweetest place

Do parent have problem and Need Check blood sugar of children during the night?



Alternative Solutions



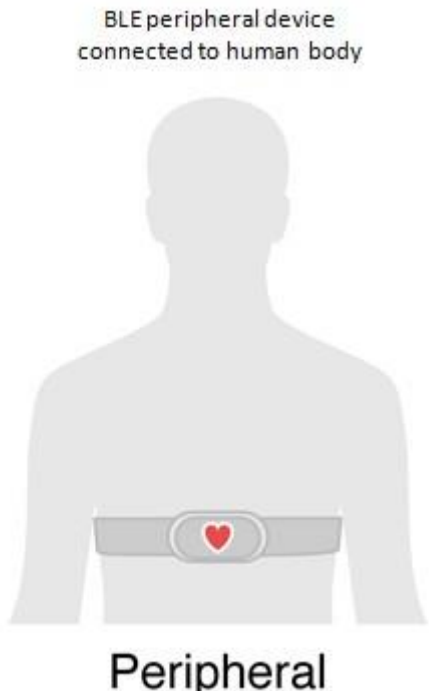
Problem Validation with Doctor

Dr. Asad Allah Fath Allahpour
Super Pediatrician
Faholapour@yahoo.com

problem

hypoglycemia speeds one’s heart rate and alters heart rate variability, which is the normal beat-to-beat variation in heartbeats.
using a Heart rate wearable device in an outpatient setting seems promising for alerting to upcoming hypoglycemia.

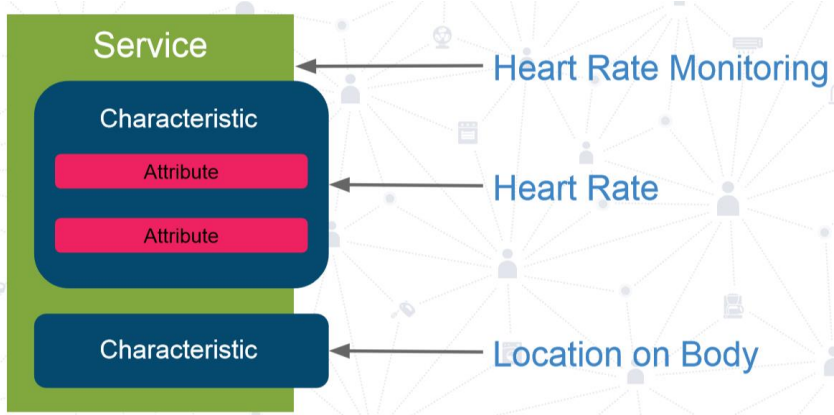
Architecture and Technology



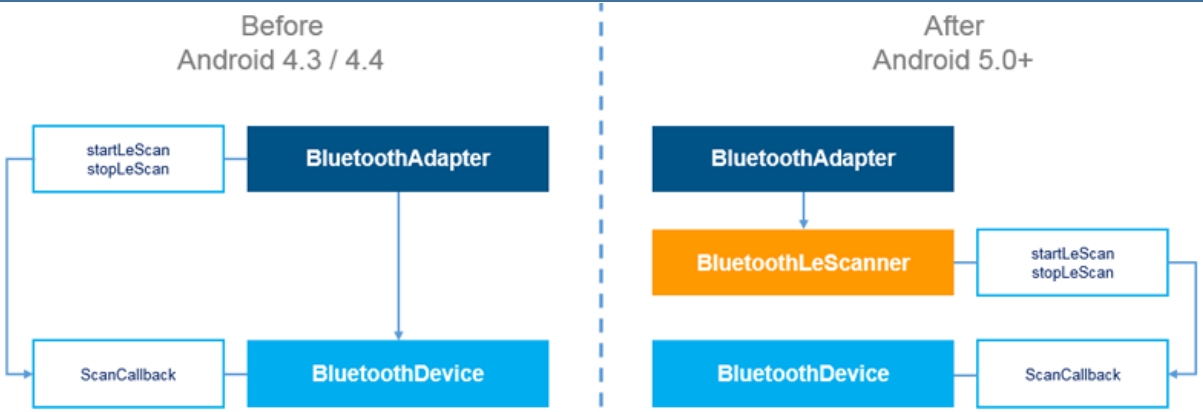
Bluetooth 4.0

Low Energy
50 meter range
1 Mbt/s transfer

consumes 50-99% less power than Bluetooth Classic
(depending on the use case)



Architecture and Technology




```
BluetoothManager manager = (BluetoothManager) getSystemService(Context.BLUETOOTH_SERVICE);
adapter = manager.getAdapter();
boolean started = adapter.startLeScan(mLeScanCallback);
```

```
BluetoothAdapter.LeScanCallback mLeScanCallback = new BluetoothAdapter.LeScanCallback() {
    @Override
    public void onLeScan(final BluetoothDevice device, int rssi, final byte[] scanRecord) {

        //  gestione device…

    }
};
```

Architecture and Technology

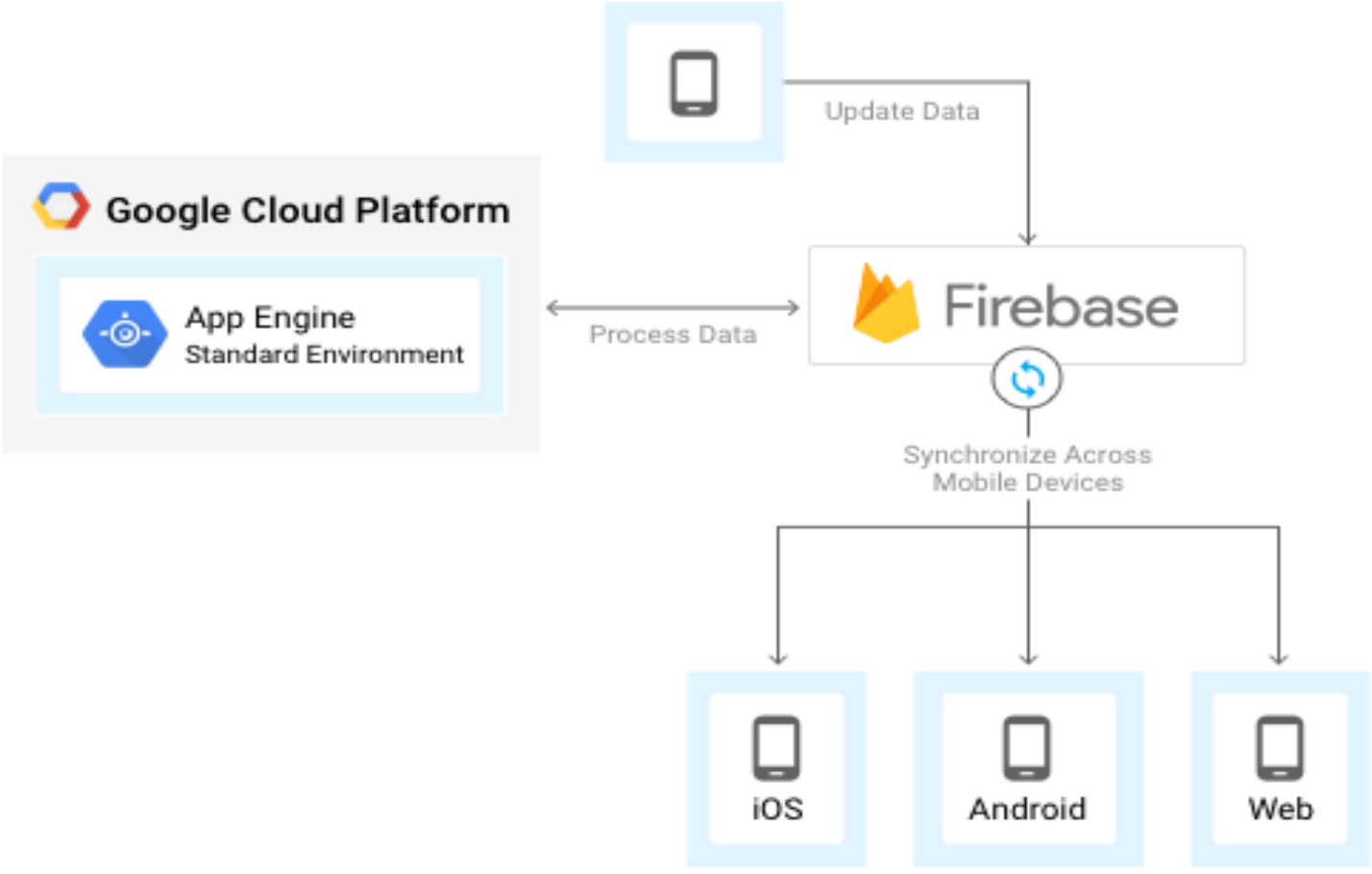

```
BluetoothGatt mGatt = device.connectGatt(getActivity(), true, new BluetoothGattCallback() {
    String SERVICE = "0000180F-0000-1000-8000-00805F9B34FB";
    String LEVEL =   "00002A19-0000-1000-8000-00805F9B34FB";

    @Override
    public void onConnectionStateChange(BluetoothGatt gatt, int status, int newState) {
        if (status == BluetoothGatt.GATT_SUCCESS && newState == BluetoothGatt.STATE_CONNECTED) {
            BluetoothGattService service = gatt.getService(UUID.fromString(SERVICE));
            BluetoothGattCharacteristic ch = service.getCharacteristics(UUID.fromString(LEVEL));
            gatt.readCharacteristic(ch);
        }
    }
    @Override
    public void onCharacteristicRead(BluetoothGatt gatt, BluetoothGattCharacteristic ch, int st) {
        final int batteryLevel = ch.getIntValue(BluetoothGattCharacteristic.FORMAT_UINT8, 0);

    }
});
```

2

3

Architecture and Technology

DOC HERO
make the world a sweetest place



Sign in with Google

Sign in with Facebook

Bitbucket

LE

Android Studio

Code sample


```
implementation "com.google.android.gms:play-services-auth:11.0.4"
implementation "com.google.android.gms:play-services-base:11.0.4"

implementation 'com.google.firebase:firebase-core:11.0.4'
implementation 'com.google.firebase:firebase-auth:11.0.4'
implementation('com.firebaseui:firebase-ui-auth:2.2.0') {
    exclude group: 'com.google.android.gms'
    exclude group: 'com.google.firebase'
}
compile "com.google.firebase:firebase-database:11.0.4"
compile "com.google.firebase:firebase-storage:11.0.4"
implementation 'com.firebaseui:firebase-ui-database:2.2.0'
//implementation 'com.firebaseui:firebase-ui-storage:2.2.0'

compile 'com.github.bumptech.glide:glide:3.8.0'

//compile 'com.android.support:support-annotations:24.2.0'
implementation 'com.github.PhilJay:MPAndroidChart:v3.0.3'
```

Firebase Real Time Database

- Real time
- Offline
- Accessible

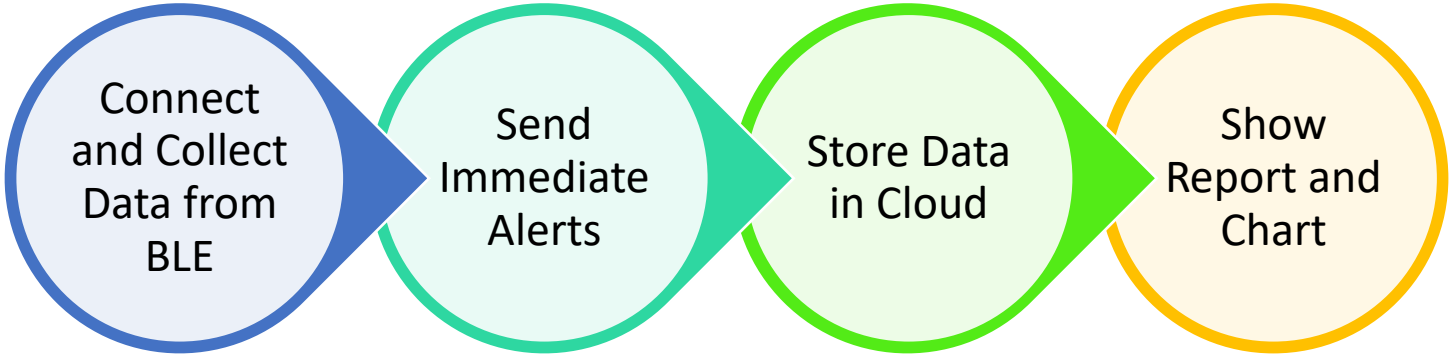

```
public void mgetAttendantOnce(String ParentID, final AutomaticPlanActivity.OnGetDataListener listener) {
    listener.onStart();

    DatabaseReference _ref = FirebaseDatabase.getInstance().getReference( s: "AttendantModel").child(ParentID);
    _ref.keepSynced( b: true);
    Query myquery = _ref.orderByChild("name");

    myquery.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(DataSnapshot dataSnapshot) {
            listener.onSuccess(dataSnapshot);
        }

        @Override
        public void onCancelled(DatabaseError databaseError) {
            listener.onFailed(databaseError);
        }
    });
}
```

Code sample

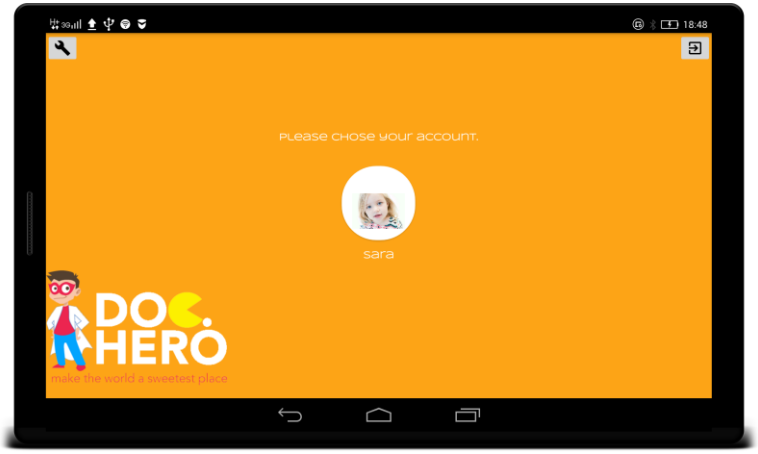

```
//Config Firebase Authentication Listener
mAuthStateListener = new FirebaseAuth.AuthStateListener() {
    @Override
    public void onAuthStateChanged(@NonNull FirebaseAuth firebaseAuth) {
        FirebaseUser user = firebaseAuth.getCurrentUser();
        if (user != null) {
            // Toast.makeText(MainActivity.this, "You are now signed in. welcome to a
            onSignedInInitialize(user.getDisplayName(), user.getUid());
        } else {
            // Choose authentication providers
            List<AuthUI.IdpConfig> providers = Arrays.asList(
                    new AuthUI.IdpConfig.Builder(AuthUI.EMAIL_PROVIDER).build(),
                    //new AuthUI.IdpConfig.Builder(AuthUI.PHONE_VERIFICATION_PROVIDER
                    new AuthUI.IdpConfig.Builder(AuthUI.GOOGLE_PROVIDER).build()
                    //new AuthUI.IdpConfig.Builder(AuthUI.FACEBOOK_PROVIDER).build(),
                    //new AuthUI.IdpConfig.Builder(AuthUI.TWITTER_PROVIDER).build()
            );
            onSignedOutInitialize();
            // Create and launch sign-in intent
            startActivityForResult(
                    AuthUI.getInstance()
                            .createSignInIntentBuilder()
                            .setIsSmartLockEnabled(false)
                            .setAvailableProviders(providers)
                            .setLogo(R.drawable.common_google_signin_btn_icon_dark)
                            .build(),
                    RC_SIGN_IN);
```

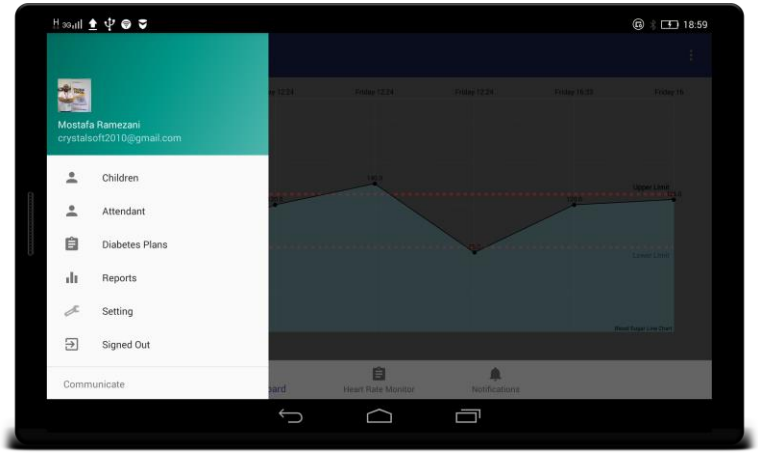

```
<uses-permission android:name="android.permission.CALL_PHONE" />
<uses-permission android:name="android.permission.SEND_SMS" />
<uses-permission android:name="android.permission.BLUETOOTH" />
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />
<uses-permission android:name="android.permission.CHANGE_WIFI_MULTICAST_STATE" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.WAKE_LOCK" />
<uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
<uses-permission android:name="com.android.alarm.permission.SET_ALARM" />
<uses-permission android:name="android.permission.GET_TASKS" />
```

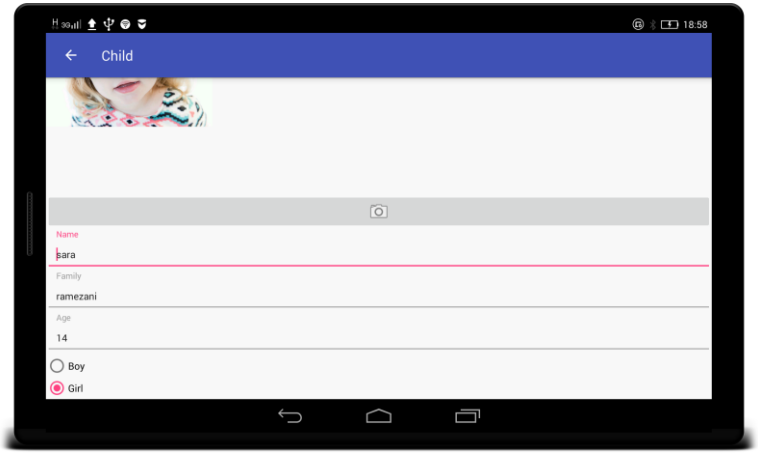
Road Map

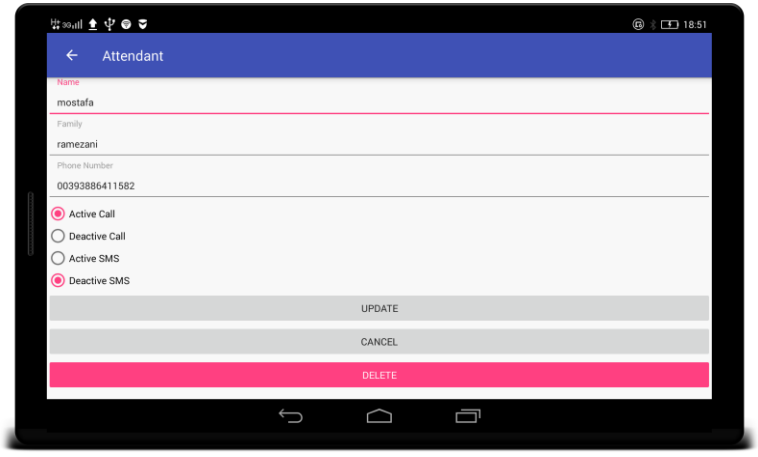

Connect and Collect Data from BLE
Send Immediate Alerts
Store Data in Cloud
Show Report and Chart

Android application and Screenshot





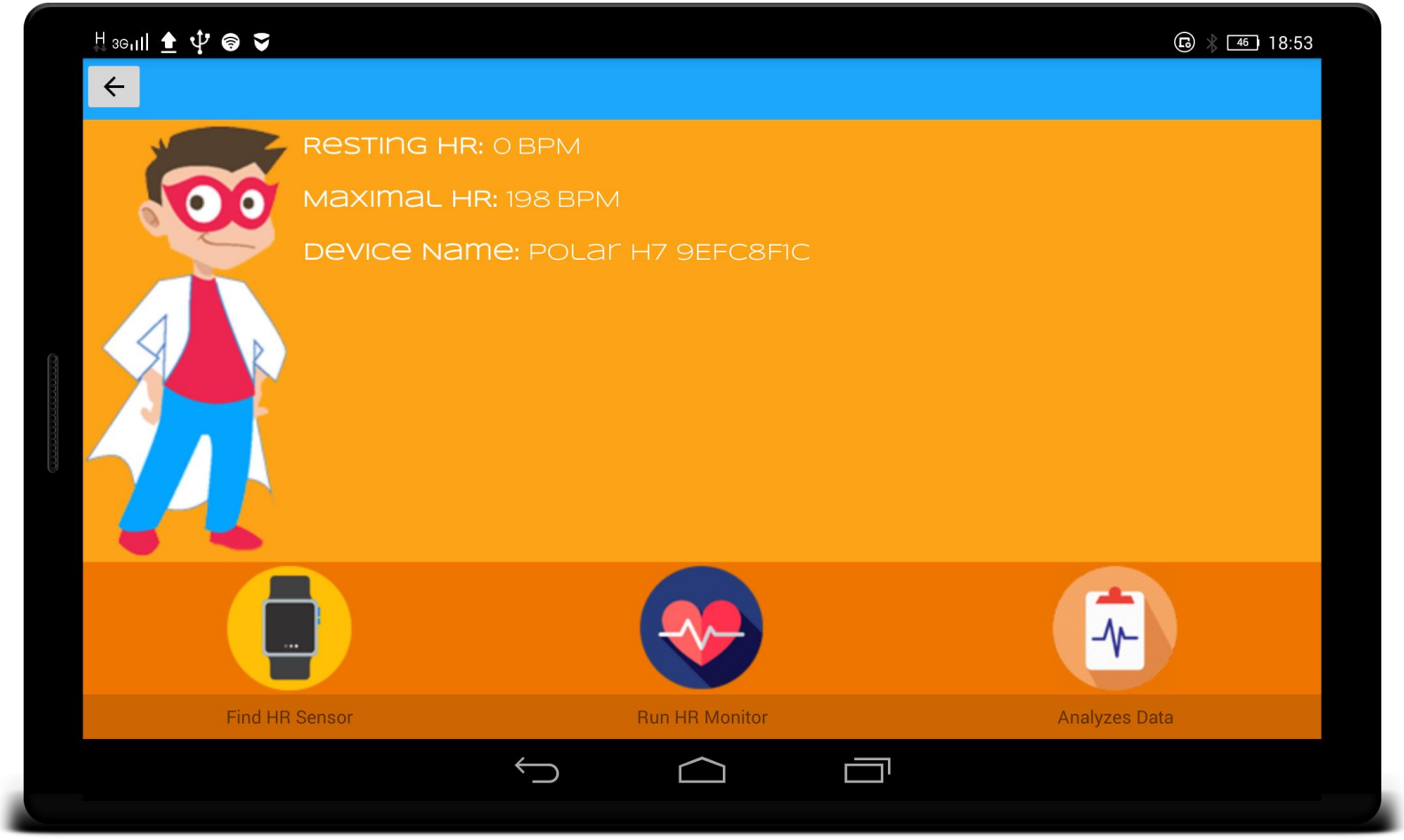




Initial Application such as Add Children or Add Attendants

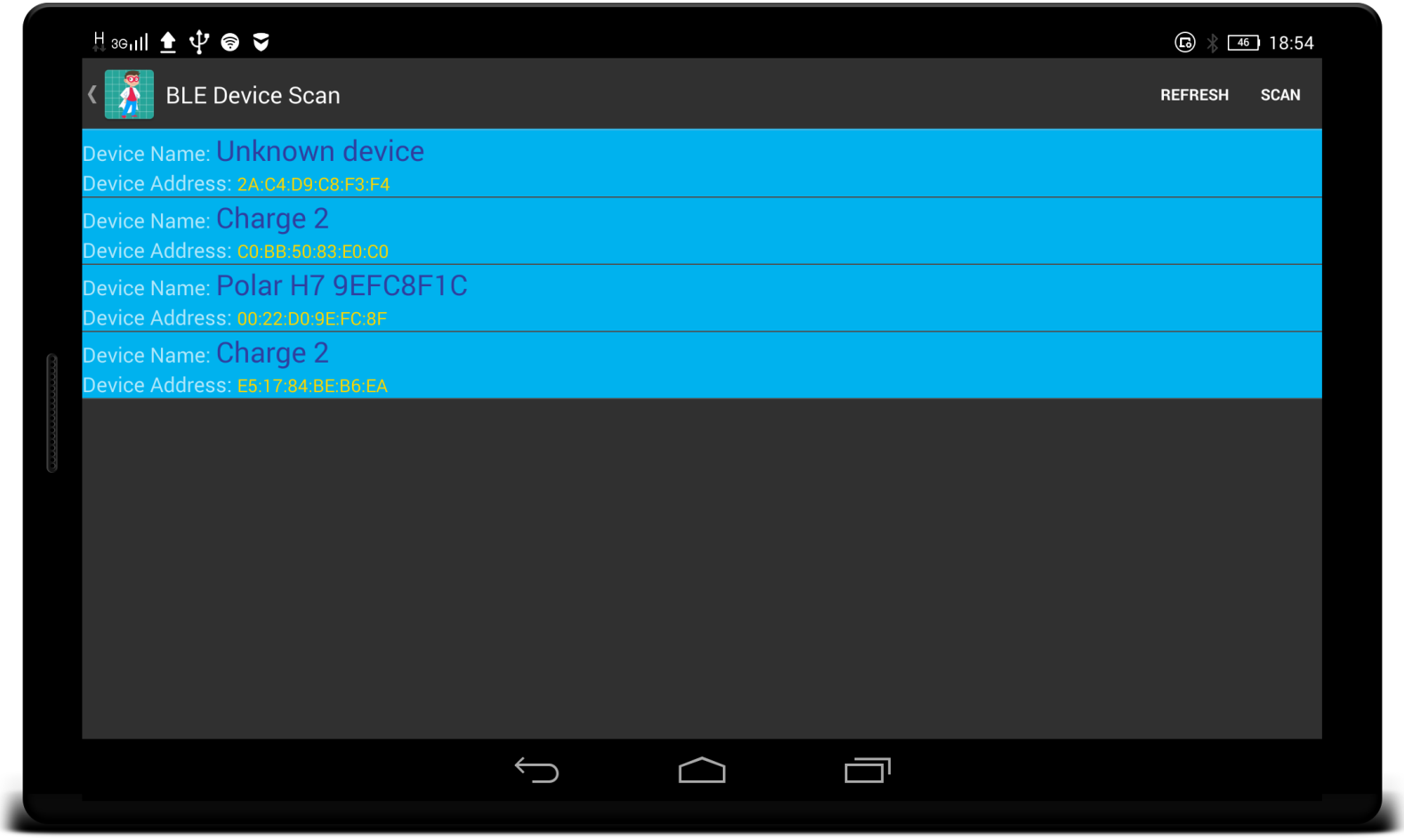
Android application and Screenshot

DOC HERO
make the world a sweetest place



Dashboard of Heart Rate Monitoring

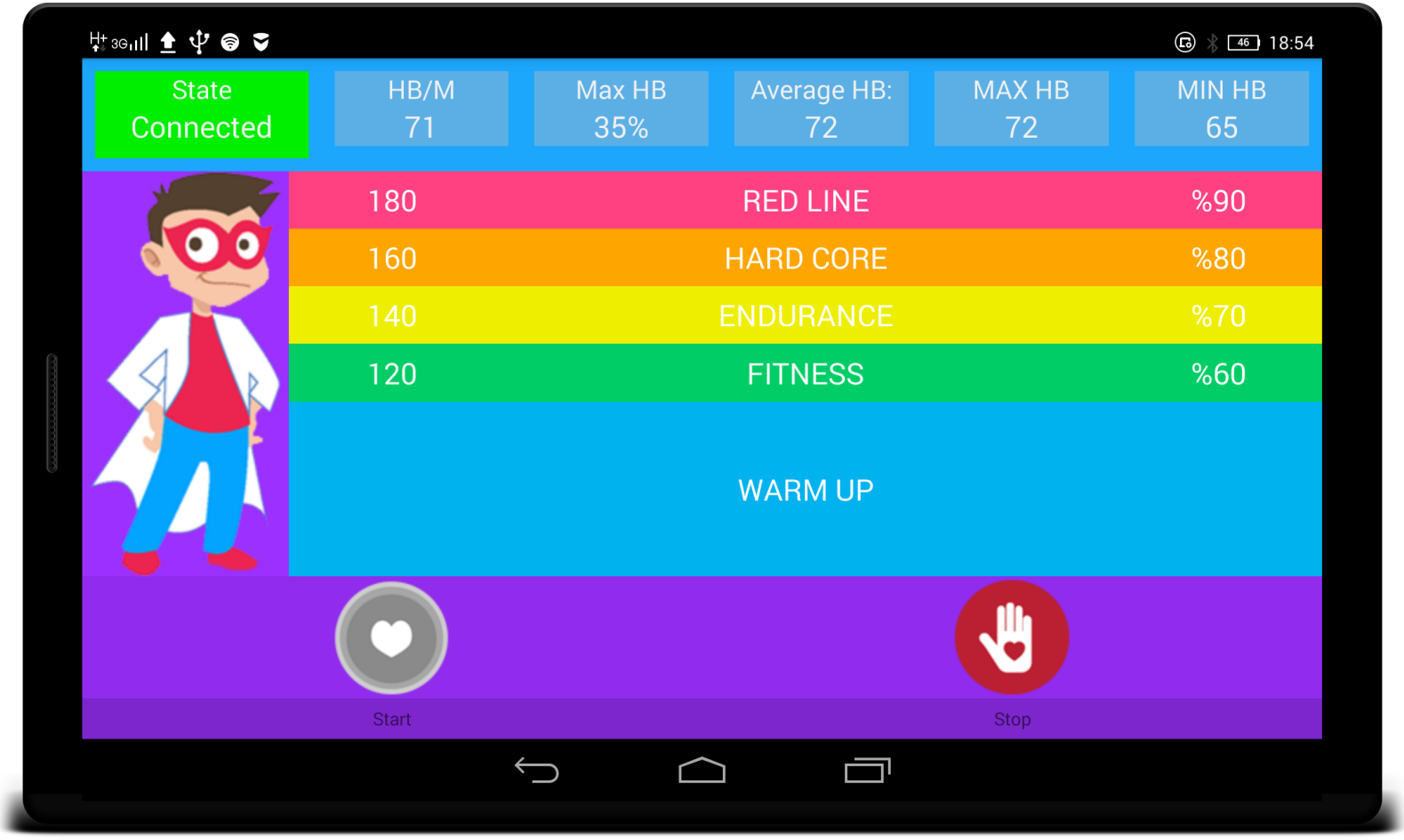
Android application and Screenshot



Scan BLE Device and Connect to Device

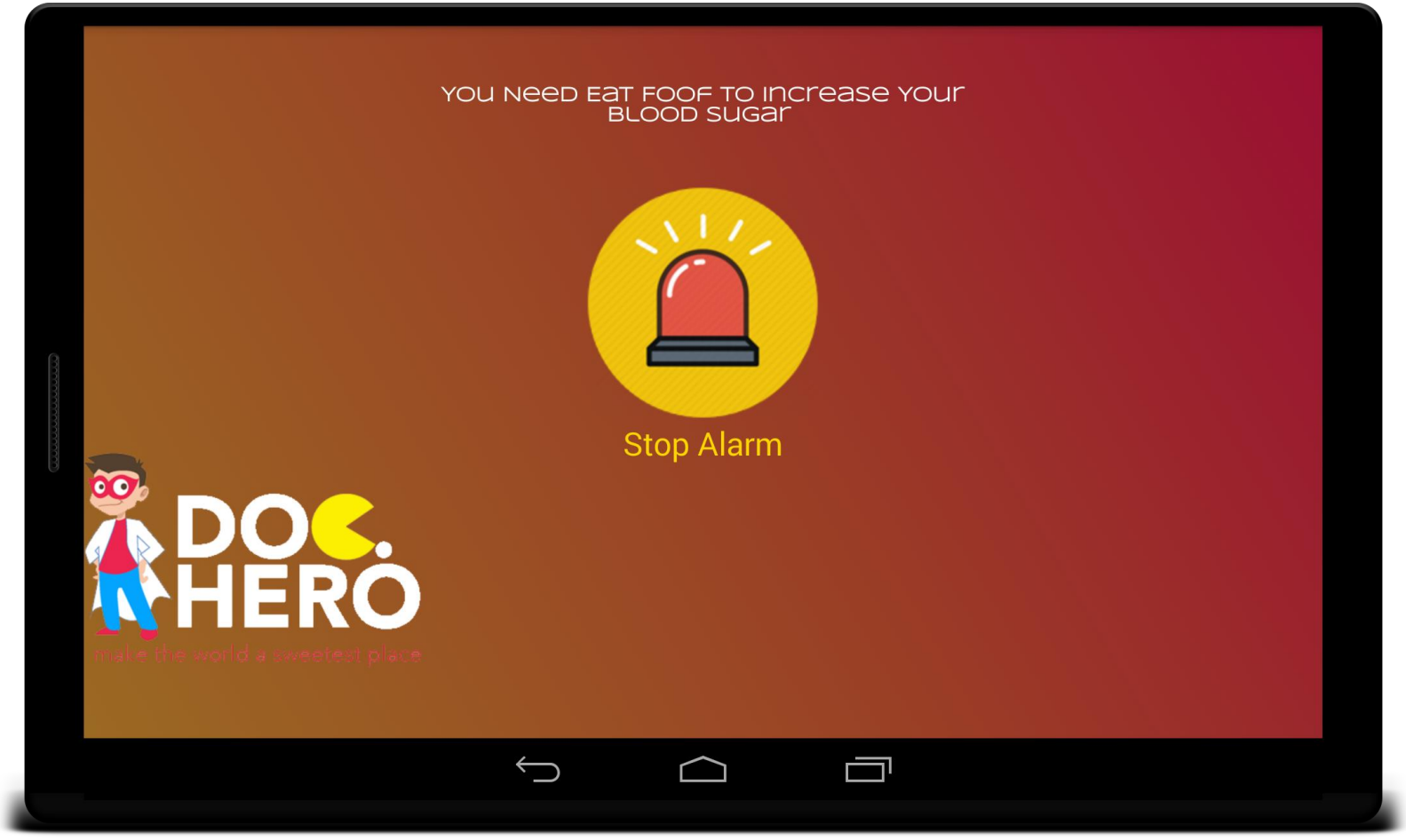
Android application and Screenshot

DOC HERO
make the world a sweetest place



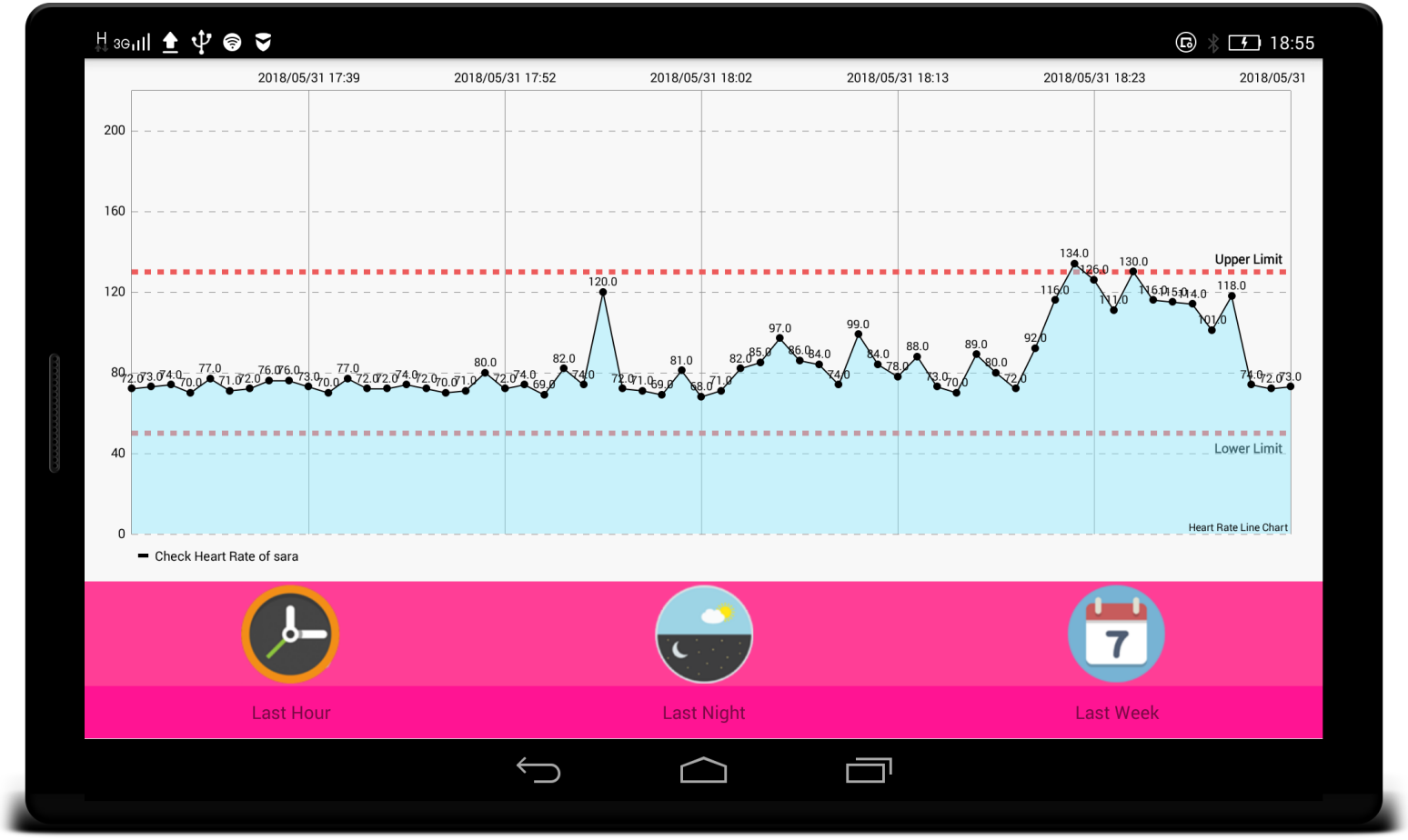
Start Heart Rate Monitoring

Android application and Screenshot



Run Alarm When Heart Rate Reach To Dangerous Situation

Android application and Screenshot



Different Report based on Time Duration

User Evaluation

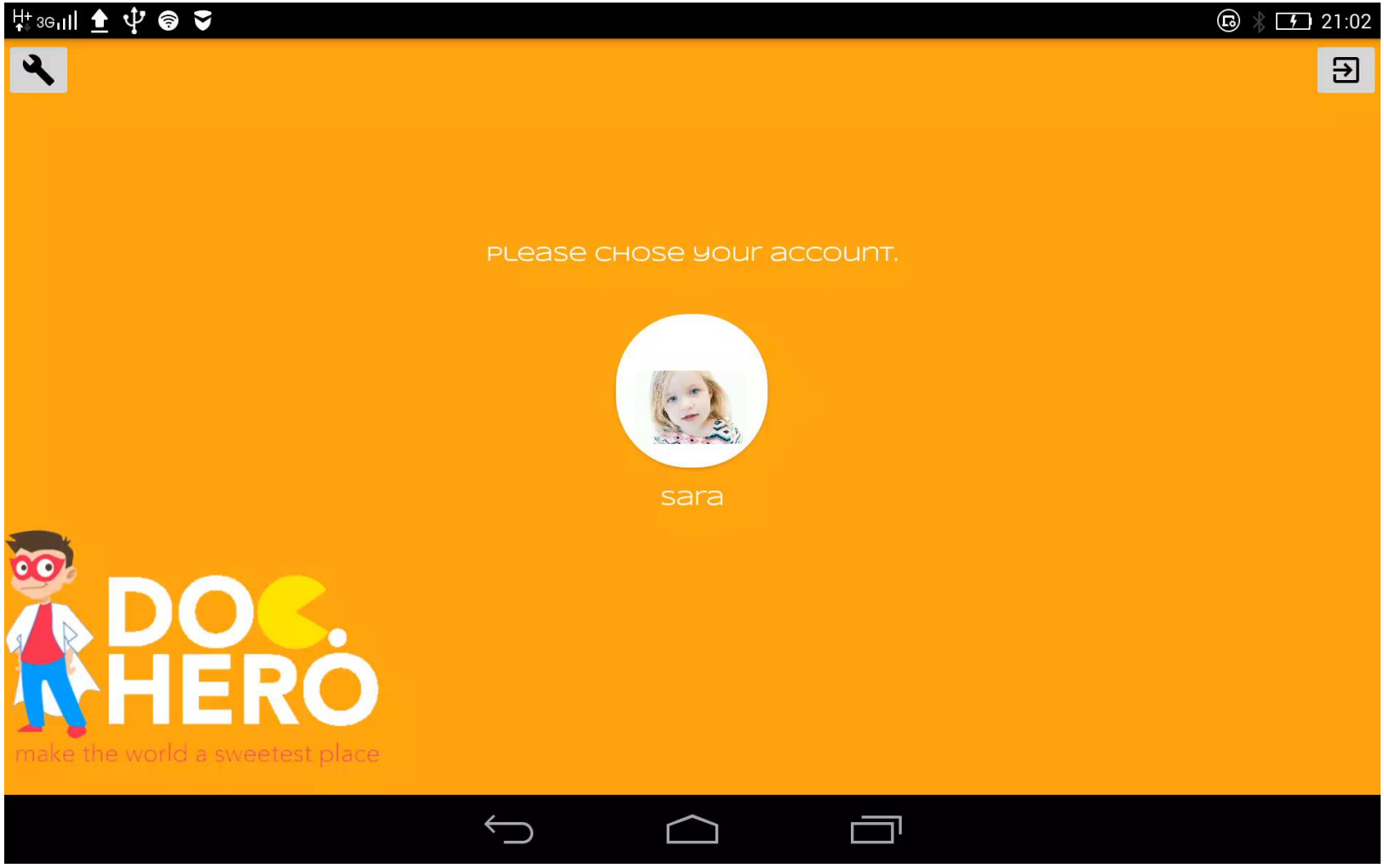
SUMMARY

Every user succeeded in completing the tasks and they simply learned and used application.

Demo App

Initial App and Use Heart Rate Monitoring

Demo App



Run Alarm of High Heart Rate

The End

Thank you
for
your attention !

https://fanoos.github.io/heart_rate_monitor/

https://github.com/fanoos/heart_rate_monitor/tree/master/Download%20App

https://github.com/fanoos/heart_rate_monitor