# Heart Rate Monitor Sample App Development

Contents

[Heart Rate Monitor Sample App Development 1](#_Toc479588273)

[Heart Rate Monitor Sample App Development using Tizen Studio 3](#_Toc479588274)

[Pre-Requisite: 3](#_Toc479588275)

[Create New Tizen Project 3](#_Toc479588276)

[Modify config.xml 7](#_Toc479588277)

[Add image resources 7](#_Toc479588278)

[Modify index.html 8](#_Toc479588279)

[Add JavaScript code 9](#_Toc479588280)

[Add styles to HTML elements 15](#_Toc479588281)

[Running application 16](#_Toc479588282)

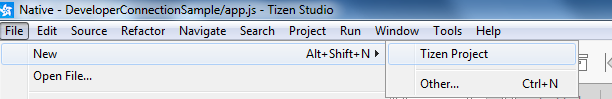
# Heart Rate Monitor Sample App Development using Tizen Studio 1.1.1

## Pre-Requisite:

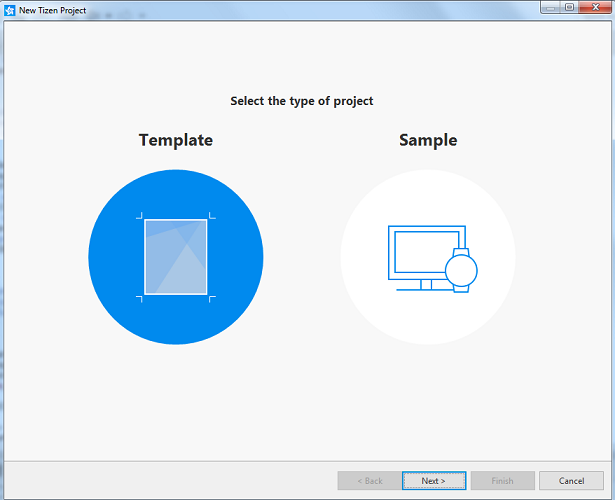
1. PC with Tizen SDK Setup
2. Gear S2/S3 device

## Create New Tizen Project

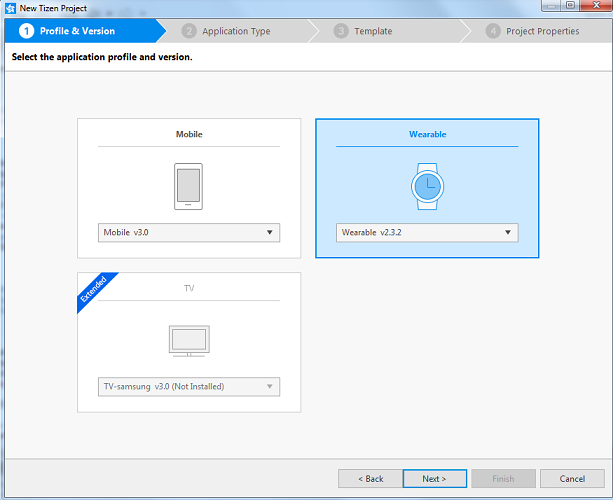
1. Click File -> New -> Tize Project



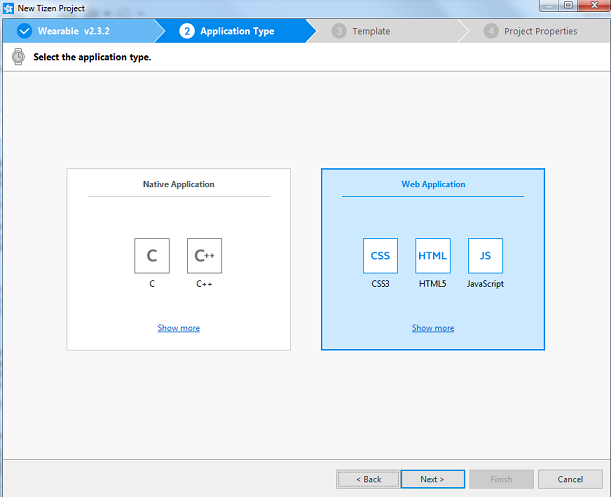
1. Select Template & click Next



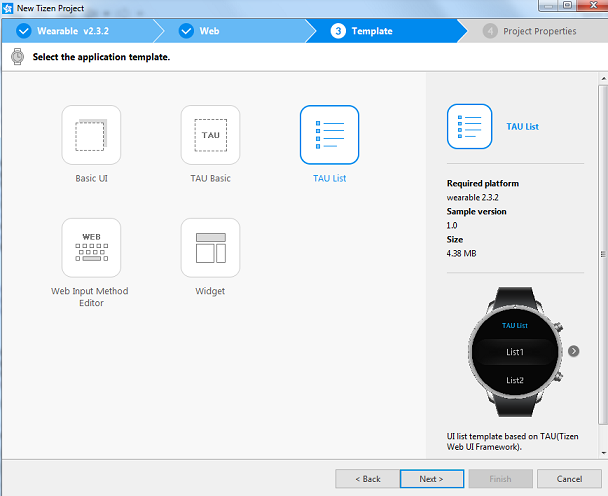
1. Select Wearable2.3.2 & click Next



1. Select Web Application & click Next.



1. Select TAU List & click Next.



1. Change the Project Name

## Modify config.xml

1. Open config.xml from project explorer.
2. Add privilege to read heart rate data

*<tizen:privilege name="http://tizen.org/privilege/healthinfo"/>*

1. Add privilege to launch application on connected mobile device

<tizen:privilege name=*"http://developer.samsung.com/tizen/privilege/accessoryprotocol"*/>

<tizen:privilege name=*"http://tizen.org/privilege/application.launch"*/>

***Config.xml file will look like as below after these modifications***

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<widget xmlns=*"http://www.w3.org/ns/widgets"* xmlns:tizen=*"http://tizen.org/ns/widgets"* id=*"http://yourdomain/DeveloperConnectionSample"* version=*"1.0.0"* viewmodes=*"maximized"*>

<tizen:application id=*"8JNTWwwj8W.DeveloperConnectionSample"* package=*"8JNTWwwj8W"* required\_version=*"2.3.2"*/>

<content src=*"index.html"*/>

<feature name=*"http://tizen.org/feature/screen.size.normal"*/>

<icon src=*"icon.png"*/>

<name>Workshop</name>

<!--healthinfo permission is required to read the heart rate data -->

<tizen:privilege name=*"http://tizen.org/privilege/healthinfo"*/>

<!-- Below 2 permssion required for deep linking http://img-developer.samsung.com/onlinedocs/gear/Extension\_PG\_Web/html/prog\_guide/remote\_ww.htm-->

<tizen:privilege name=*"http://developer.samsung.com/tizen/privilege/accessoryprotocol"*/>

<tizen:privilege name=*"http://tizen.org/privilege/application.launch"*/>

<tizen:privilege name=*"http://tizen.org/privilege/healthinfo"*/>

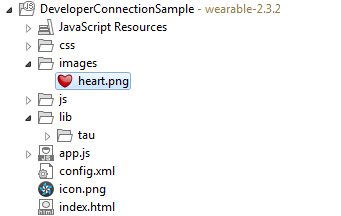
<tizen:profile name=*"wearable"*/>

<tizen:setting hwkey-event=*"enable"*/>

</widget>

## Add image resources

1. Create new folder in project and copy heart.png in it.

## Modify index.html

1. Open index.html from project explorer and modify index.html as below

<!DOCTYPE html>

<html>

<head>

<meta name=*"viewport"* content=*"width=device-width,user-scalable=no"*>

<title>Heart Rate Monitor</title>

<!-- \*\*\*\*\*\*\*\* Loading style css file of the application \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* -->

<link rel=*"stylesheet"* href=*"lib/tau/wearable/theme/default/tau.min.css"*>

<link rel=*"stylesheet"* media=*"all and (-tizen-geometric-shape: circle)"*

href=*"lib/tau/wearable/theme/default/tau.circle.min.css"*>

<link rel=*"stylesheet"* href=*"css/style.css"*>

</head>

<body onload="init()">

<!-- \*\*\*\*\*\*\*\* Launch page of the applicatin\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* -->

<div class=*"ui-page ui-page-active"* id=*"main"*>

<div class=*"ui-header"*>

<h2 class=*"ui-title"*>Heart Rate Monitor</h2>

</div>

<div class=*"ui-content"*>

<div id=*"heart-rate-value"*></div>

<div id=*"heart"*>

<div id=*"heart-img"*></div>

</div>

</div>

<footer class=*"ui-footer ui-bottom-button"*>

<a id=*"hrm-control-btn"* class=*"ui-btn"*>Start</a>

</footer>

</div>

<!--\*\*\*\*\* Below page will be shown when Stop button is tapped by the user\*\*\*\*\*\* -->

<div class=*"ui-page"* id=*"info"*>

<div class=*"ui-header"*>

<h2 class=*"ui-title"*>HRM Info</h2>

</div>

<div class=*"ui-content"*>

<div class=*"info-text"*>

not enough info recorded.

</div>

</div>

<footer class=*"ui-footer ui-bottom-button"*>

<a id=*"info-back-btn"* class=*"ui-btn"*>Back</a>

</footer>

</div>

<!-- \*\*\*\*\*\* Below page will be show the user heart rate information\*\*\*\*\*\*\*\*\*\*\*\* -->

<div class=*"ui-page"* id=*"information"*>

<header class=*"ui-header"*>

<h2 class=*"ui-title"*>HRM Information</h2>

</header>

<div class=*"ui-content"*>

<ul class=*"ui-listview"*>

<li id =*"currentHR"*>Current HR</li>

<li id =*"maxHR"*>Max HR</li>

<li id =*"minHR"*>Min HR</li>

<li id =*"avgHR"*>Avg HR</li>

</ul>

</div>

<!--\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Button to deeplink to ABC Mouse application \*\*\*\*\*\*\*\*\*\*\* -->

<footer class=*"ui-footer ui-bottom-button"*>

<a id=*"idDeeplink"* class=*"ui-btn"*>On Phone</a>

</footer>

</div>

<!-- \*\*\*\*\*\*\*\* Loading javascript file of the application \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* -->

<script type=*"text/javascript"* src=*"lib/tau/wearable/js/tau.min.js"*></script>

<script type=*"text/javascript"* src=*"js/circle-helper.js"*></script>

<script src=*"app.js"*></script>

<script src=*"lowBatteryCheck.js"*></script>

</body>

</html>

## Add JavaScript code

1. Open app.js from project explorer and add application logic as below.

/\*\*\*\* Global variable used in the application \*\*\*\*/

**var** TEXT\_STOP = 'Stop';

**var** TEXT\_START = 'Start';

**var** heartRateEl;

**var** heartImg;

**var** infoBackBtn;

**var** hrmControlBtn;

**var** deeplinkBtn;

**var** sensorStarted = **false**;

/\*\*TAG for application logs \*\*\*\*\*/

**var** TAG = "HeartRateMonitor \*\*\*\*\*";

/\*\*

\* function: init()

\* This function will be called on "onload" event of the html

\* This is the very fist function which will be called when the app is loaded.

\* Initialization of HTML variables and binding of click events is done inside this function.

\*

\*/

**function** init() {

console.log(TAG+ "init() function called");

/\*\*\* Initialize the html variables \*\*\*\*\*/

heartRateEl = document.getElementById('heart-rate-value');

heartImg = document.getElementById('heart-img');

infoBackBtn = document.getElementById('info-back-btn');

hrmControlBtn= document.getElementById('hrm-control-btn');

deeplinkBtn = document.getElementById('idDeeplink');

/\*\*\* Registering click event handler for buttons \*/

hrmControlBtn.addEventListener('click', onhrmControlBtnClick);

infoBackBtn.addEventListener('click', onInfoBackBtnClick);

deeplinkBtn.addEventListener('click', onDeeplinkToMobile);

}

/\*\*

\* function: onhrmControlBtnClick()

\* This function is a click event handler for HRM sensor Start/Stop button

\* Toggles the sensor state

\* Toggles the text of the button

\*/

**function** onhrmControlBtnClick() {

console.log(TAG+"onhrmControlBtnClick() called...");

**if** (hrmControlBtn.innerHTML === TEXT\_START){

console.log("info on button = start");

startSensor();

} **else** {

console.log("info on button = stop");

stopSensor();

showMeasuringInfo();

}

}

/\*\*

\* function: startSensor()

\* Starts the HRM sensor and registers a change listener.

\* Update the UI: change button text to Stop.

\*/

**function** startSensor() {

console.log(TAG+"start sensor() called...");

sensorStarted = **true**;

hrmControlBtn.innerHTML = TEXT\_STOP;

/\*\* Start heart image animation, 'animate' class is defined in style.css \*/

**if** (!heartImg.classList.contains('animate')) {

heartImg.classList.add('animate');

}

/\*\* Start the HRM sensor\*/

tizen.humanactivitymonitor.start('HRM', onHeartRateDataChange, onerrorCB);

}

/\*\*

\* function: stopSensor()

\* Stops the sensor

\* Update the UI: Hides measuring text, stop animation on heart image and change button text to Start.

\*/

**function** stopSensor() {

console.log(TAG+"stopSensor() called...");

sensorStarted = **false**;

/\*\*Remove animation class from heart icon \*/

heartImg.classList.remove('animate');

/\*\*\* Stop sensor \*\*\*/

tizen.humanactivitymonitor.stop("HRM");

/\*\*\* Change button text to start \*\*\*/

hrmControlBtn.innerHTML = TEXT\_START;

}

/\*\*

\* function: onHeartRateDataChange()

\* This is a callback function which will be called after receiving the current heart rate of the user

\* All the calucations for setting up the min, max & avg heart rate being done inside this function.

\*

\*/

**function** onHeartRateDataChange(heartRateInfo) {

console.log(TAG+"onHeartRateDataChange() called...");

**if** (!sensorStarted){

**return**;

}

heartRateEl.innerHTML = '';

**var** rate = heartRateInfo.heartRate;

**var** activePage = document.getElementsByClassName('ui-page-active')[0];

**var** activePageId = activePage ? activePage.id : '';

/\*\*

\* If heart rate value is valid, set the value and show the information page.

\*/

**if** (rate > 1) {

console.log(TAG+"heartRateEl is valid information...");

/\*Show the information page & set the heart rate value on Heart Rate image\*/

tau.changePage('#information');

heartRateEl.innerHTML = rate;

/\*\*

\* Sensor only provide the current heart rate of the user, to show minimum heart rate, maximum heart rate & average heart rate

\* We have to do the calculations at application side. We are using tizen pereferences to achieve this.

\* https://developer.tizen.org/ko/development/guides/web-application/application-management/application-preferences?langredirect=1

\* tizen pereferences are something like we have shared preferences in Abdroid.

\*/

document.getElementById("currentHR").innerHTML = "Current HR: "+rate;

/\*\* MINHR pereferences Key is used to store the minimum heart rate

\* If MINHR key is alreadty exists then get the value of the key and if saved value is greater than the current Heart rate then update the key with the current heart rate

\* If MINHR key does not exists then it means this is very first heart rate reading, then create the MINHR key and save the value as the current heart rate

\*\*/

**if** (tizen.preference.exists("MINHR")) {

**var** value = tizen.preference.getValue("MINHR");

**if**(value > rate) {

tizen.preference.setValue("MINHR", rate);

}

} **else** {

tizen.preference.setValue("MINHR", rate);

}

/\*\* MAXHR pereferences Key is used to store the maximum heart rate

\*If MAXHR key is alreadty exists then get the value of the key and if saved value is lesser than the current Heart rate then update the key with the current heart rate

\*If MAXHR key does not exists then it means this is very first heart rate reading, then create the MAXHR key and save the value as the current heart rate

\*\*/

**if** (tizen.preference.exists("MAXHR")) {

**var** value = tizen.preference.getValue("MAXHR");

**if**(value < rate) {

tizen.preference.setValue("MAXHR", rate);

}

} **else** {

tizen.preference.setValue("MAXHR", rate);

}

/\*\* AVGHR Key is used to store the maximum heart rate & COUNT key is used to store the number of successfull heart rate reading,

\* COUNT will be used in calculating the average heart rate of the user

\* If AVGHR & COUNT key are alreadty exists then get the value of the keys and perform the calculations.

\* If AVGHR & COUNT keys does not exists then it means this is very first heart rate reading, then create AVGHR key and save the value as the current heart rate

\* & set the value of COUNT key as one as this is the first hear rate reading.

\*\*/

**if** (tizen.preference.exists("AVGHR") && tizen.preference.exists("COUNT")) {

**var** value = tizen.preference.getValue("AVGHR");

**var** count = tizen.preference.getValue("COUNT");

**var** total = value \* count + rate;

count = count + 1;

**var** avg = total/count;

tizen.preference.setValue("AVGHR", avg);

tizen.preference.setValue("COUNT", count);

} **else** {

tizen.preference.setValue("AVGHR", rate);

tizen.preference.setValue("COUNT", 1);

}

/\*\* Get the values of the heart rate from perefernce keys and show that on html page \*/

**var** minHR = tizen.preference.getValue("MINHR");

**var** maxHR = tizen.preference.getValue("MAXHR");

/\*\* Remove decimal value from Avg HR \*/

**var** avgHR = parseInt(tizen.preference.getValue("AVGHR"));

document.getElementById("minHR").innerHTML = "Min HR: "+ minHR;

document.getElementById("maxHR").innerHTML = "Max HR: "+ maxHR;

document.getElementById("avgHR").innerHTML = "Avg HR: "+ avgHR;

console.log(TAG + "Current HR: "+rate);

console.log(TAG + "Minimum HR: "+minHR);

console.log(TAG + "Maximum HR: "+maxHR);

console.log(TAG + "Average HR: "+avgHR);

/\*\* Stop the sensor as we have got the hear rate reading.\*/

stopSensor();

}

}

/\*\*

\* function: onerrorCB()

\* Call back when an error occurs in starting the HRM sensor

\*\*/

**function** onerrorCB(error) {

console.log(TAG+"Error name:"+error.name + ", message: "+error.message);

}

/\*\*

\* function: showMeasuringInfo()

\* Displas information popup when stop button is tapped.

\*/

**function** showMeasuringInfo() {

console.log(TAG+"showMeasuringInfo() called..");

tau.changePage('#info');

}

/\*\*

\* function: onInfoBackBtnClick()

\* Click event handler for back button on info page

\* Hides the information popup and redirects to main page.

\*/

**function** onInfoBackBtnClick() {

console.log(TAG+"onInfoBackBtnClick() called...");

tau.changePage('#main');

}

/\*\*

\* function: onDeeplinkToMobile()

\* This function will execute when user press on "On Phone" button which will be visible on HRM Information page.

\* The deeplinking is being done to one of the samsung app "ABC mouse".

\* There can be two case, once ABC mouse app is installed on the connected android mobile device and another case is app is not installed on the device.

\* If "ABC mouse" app is not installed on connected anroid mobile device => On tap of "On Phone" button, it will show the "ABC mouse" app detail page on Samsung Galaxy app store.

\* If "ABC mouse" app is already insatalled on connected anroid mobile device => => On tap of "On Phone" button, it will launch the "ABC mouse" app in android mobile device.

\* For deeplink implementation you may refer to http://img-developer.samsung.com/onlinedocs/gear/Extension\_PG\_Web/html/prog\_guide/remote\_ww.htm

\*/

**function** onDeeplinkToMobile(){

/\*\*\*\* Launched application sent a reply from android device \*\*\*\*/

**var** remoteAppControlReplyCallback = {

// callee returned success

onsuccess: **function**(data) {

console.log(TAG+"onsuccess "+data);

},

// callee returned failure

onfailure: **function**() {

console.log(TAG+"fail "+data);

}

}

**var** remoteAppCtrl = **new** webapis.RemoteApplicationControl("http://samsung.com/appcontrol/operation/remote/default", **null**,

"mobi.abcmouse.samsung", // pkg name of the ABC mouse application

"com.abcmouse.MainActivity", // MainActivity of the ABC mouse application

"Lauch app on connected mobile"

);

**try** {

webapis.remoteappcontrol.launchRemoteAppControl( remoteAppCtrl,

**function** () { // success callback method

console.log(TAG+ "app launch success callback");

},

**function**(e) { // error callback method

console.log(TAG+"app launch failure callback");

console.log(TAG+"app was not found on the connected android mobile device, try to launch the app detail page on Samsung Galaxy app store");

launchAppDetailPage();

},

remoteAppControlReplyCallback);

} **catch**(e) {

console.log(TAG+"Error Exception, error name : " + e.name + ", error message : " + e.message);

}

}

/\*\*

\* function: launchAppDetailPage()

\* launchAppDetailPage() will be called if ABC mosue is not installed in the connected android mobile device.

\* It will launch the app detail page on Samsung Galaxy app store.

\*/

**function** launchAppDetailPage() {

**var** remoteAppCtrl = **new** webapis.RemoteApplicationControl(

"http://samsung.com/appcontrol/operation/remote/view",

"samsungapps://ProductDetail/mobi.abcmouse.samsung/" // Create a deeplink of Samsung galaxy app store page using ABC mouse app pkg name.

);

**try** {

webapis.remoteappcontrol.launchRemoteAppControl( remoteAppCtrl,

**function** () {

console.log(TAG+ "launch success callback");

},

**function**(e) {

console.log(TAG+ "launch success callback "+e.name);

});

} **catch**(e) {

console.log(TAG+"Error Exception, error name : " + e.name + ", error message : " + e.message);

}

}

/\*\*

\* Handle the basic Back button (hard key of gear device) funtcionality

\*/

( **function** () {

window.addEventListener( 'tizenhwkey', **function**( ev ) {

**if**( ev.keyName === "back" ) {

**var** page = document.getElementsByClassName( 'ui-page-active' )[0],

pageid = page ? page.id : "";

**if**( pageid === "main" ) {

**try** {

tizen.humanactivitymonitor.stop("HRM");

tizen.application.getCurrentApplication().exit();

} **catch** (ignore) {

}

} **else** {

window.history.back();

}

}

} );

} () );

## Add styles to HTML elements

1. Open styles.css from project explorer and add following code

*.ui-title* {

font-size: *25px* *!important*;

}

*.ui-content* {

font-size: *20px*;

}

*#heart-rate-value* {

position: *absolute*;

text-align: *center*;

width: *100%*;

height: *20%*;

font-size: *17vh*;

font-weight: *bold*;

z-index: *100*;

margin-top: *15%*;

color: *#ffffff*;

}

*#heart* {

position: *absolute*;

width: *100%*;

height: *85%*;

}

*#heart-img* {

margin: *5%* *auto*;

background-image: *url(../images/heart.png)*;

background-position: *center*;

background-size: *cover*;

width: *60%*;

height: *60%*;

}

*.ui-footer* {

background-color: *transparent*;

}

*.info-text* {

text-align: *left*;

color: *#ffffff*;

font-size: *24px*;

}

@**-webkit-keyframes** **heartMove360** {

0%, 50%, 100% {

width: *35%*;

height: *35%*;

}

25%, 75% {

**width***:* 45%;

**height***:* 45%;

}

}

@media screen and (-tizen-geometric-shape: circle) {

*.ui-header* {

height: *90px*;

}

*#heart-img* {

width: *45%*;

height: *45%*;

}

*.animate* {

-webkit-animation: *heartMove360* *1.5s* *infinite*;

}

*.ui-footer* {

height: *20%*;

}

*.info-text* {

text-align: *center*;

margin-left: *5%*;

margin-right: *5%*;

}

}

## Running application

1. Right click over Project in project explorer -> Run As -> Tizen Web Application.