HYBRID APP DEV
SENSORS

DR. JAVED KHAN
v.j.khan@tue.nl khan.gr @v_j_khan

CREATIVE APPS, JAN2017

TU/e

PREVIOUS

- Practice
  - UI
  - Persistent Data
    - Localstorage
    - Firebase
  - Device Data

TODAY

- Sensors

Note

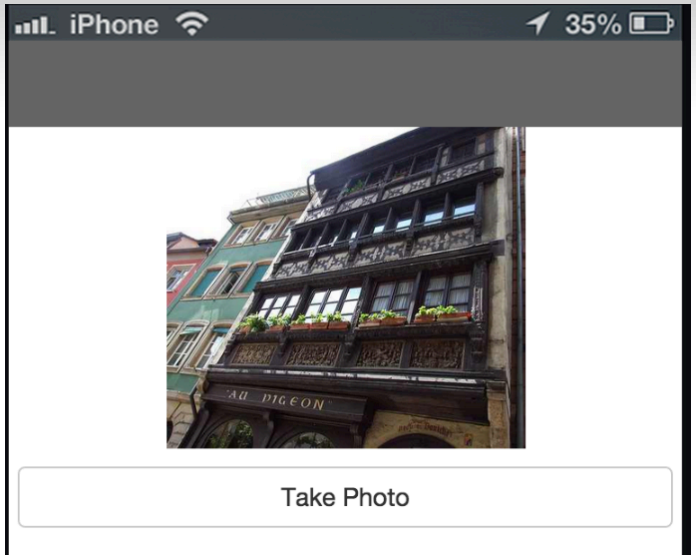
- Make sure to FIRST include the necessary plugin to your project
- Have a look at previous slides if you don’t know how to add a plugin

Plugin Types

- There are plugins that are supported by Cordova and these are stable (more mature)
  - http://cordova.apache.org/docs/en/latest/#plugin-apis
- There are plugins from “third-parties”.
  - More “exotic” functionality
  - Might not be as stable or not support all OSs
  - Need to search at: https://cordova.apache.org/plugins/

Exercise 1a: Camera

- Example:
  - UI: header, img, button
  - Interaction: when button (*id="btn"*) is clicked camera can be used to take photo, photo is then displayed on img (*id="showPhoto"*)
- Try it on your device
  - Use "Test" tab: "Push to Testing Sever" in combination with the "App Preview" app on your phone



Exercise 1a: Solution

```
function register_event_handlers()
{

    function onSuccess(imageURI) {
        var image = document.getElementById('showPhoto');
        image.src = imageURI;
    }

    function onFail(message) {
        alert('Failed because: ' + message);
    }
     /* button  #btn */
    $(document).on("click", "#btn", function(evt)
    {
        /* your code goes here */
        navigator.camera.getPicture(onSuccess, onFail, { quality: 50,
        destinationType: Camera.DestinationType.FILE_URI });
    });

    }
document.addEventListener("app.Ready", register_event_handlers, false);
```

- Adapted from: http://cordova.apache.org/docs/en/latest/reference/cordova-plugin-camera/index.html#example-

Exercise 1b: Vibration

- Add to the previous exercise a 1 second vibration when a photo is successfully taken
- Tip:
  - Look for the appropriate plugin

Exercise 1b: Solution

```
function onSuccess(imageURI) {
    var image = document.getElementById('showPhoto');
    image.src = imageURI;
    navigator.vibrate(1000);
}
```

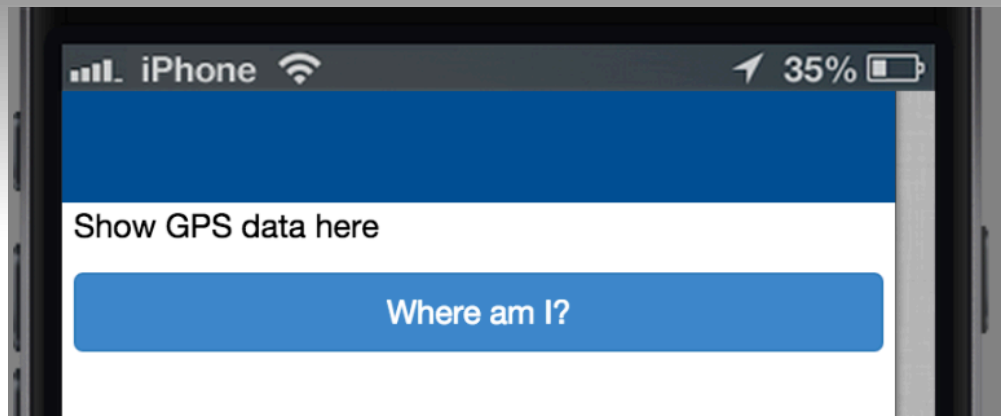
Exercise 2: Location & Maps

- Get Latitude & Longitude from sensor & use Google Maps API to show map of the user's location

Exercise 2a: Location & Maps

- When user taps on button find latitude and longitude
- Tip:
  - Emulator is handy:

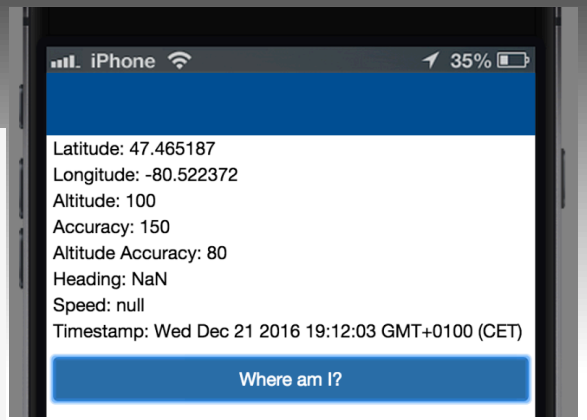
Latitude	47.465187
Longitude	-80.522372



TU/e

Exercise 2a: Solution

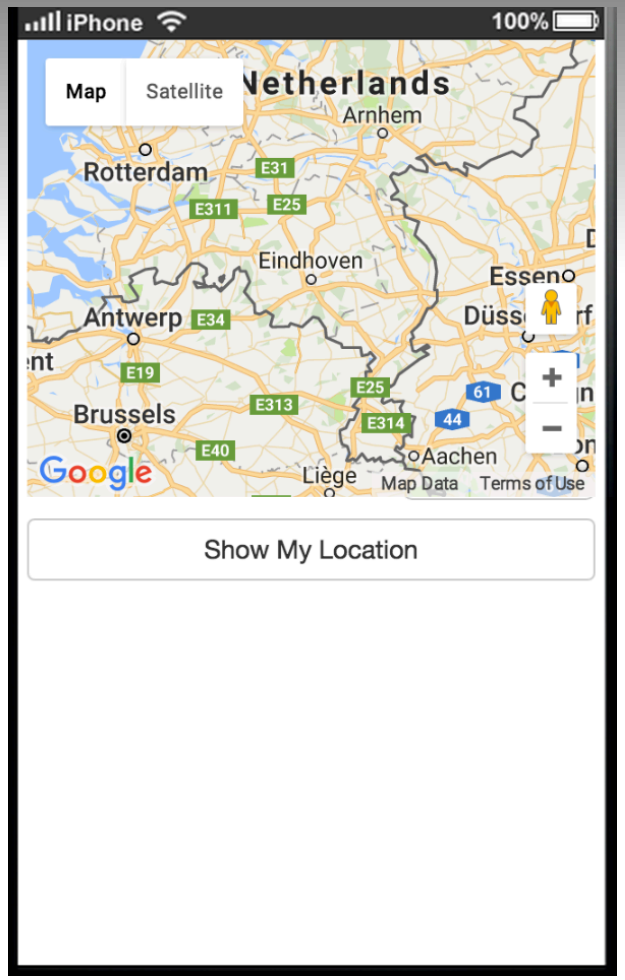
```
function register_event_handlers()
{

   var onSuccess = function(position) {
       $('#showData').html('Latitude: ' + position.coords.latitude + '<br/>' +
         'Longitude: '         + position.coords.longitude         + '<br/>' +
         'Altitude: '          + position.coords.altitude          + '<br/>' +
         'Accuracy: '          + position.coords.accuracy          + '<br/>' +
         'Altitude Accuracy: ' + position.coords.altitudeAccuracy  + '<br/>' +
         'Heading: '           + position.coords.heading           + '<br/>' +
         'Speed: '             + position.coords.speed             + '<br/>' +
         'Timestamp: '         + position.timestamp                + '<br/>');
   };

   // onError Callback receives a PositionError object
   //
   function onError(error) {
       alert('code: '    + error.code    + '\n' +
             'message: ' + error.message + '\n');
   }
    /* button  Where am I? */
   $(document).on("click", ".uib_w_3", function(evt)
   {
       /* your code goes here */
       navigator.geolocation.getCurrentPosition(onSuccess, onError);
   });

   }
document.addEventListener("app.Ready", register_event_handlers, false);
```



Adapted from: http://cordova.apache.org/docs/en/latest/reference/cordova-plugin-geolocation/index.html#example

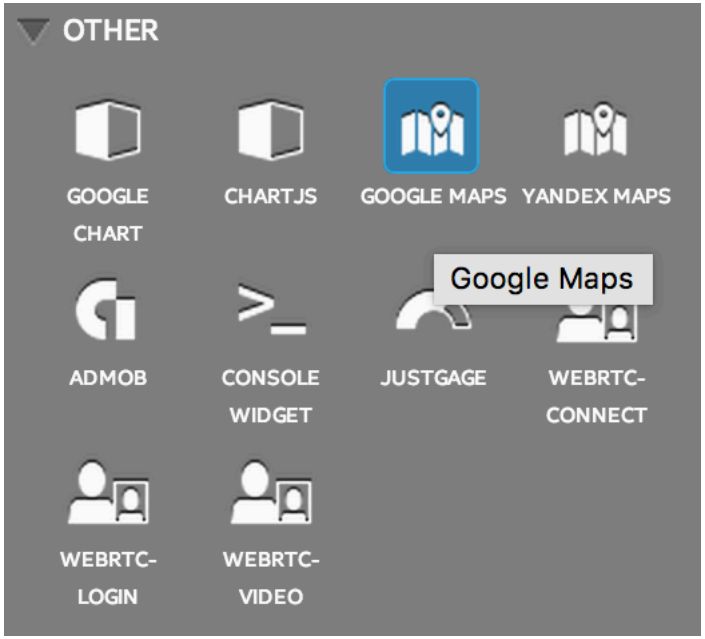
Exercise 2b: Location & Maps

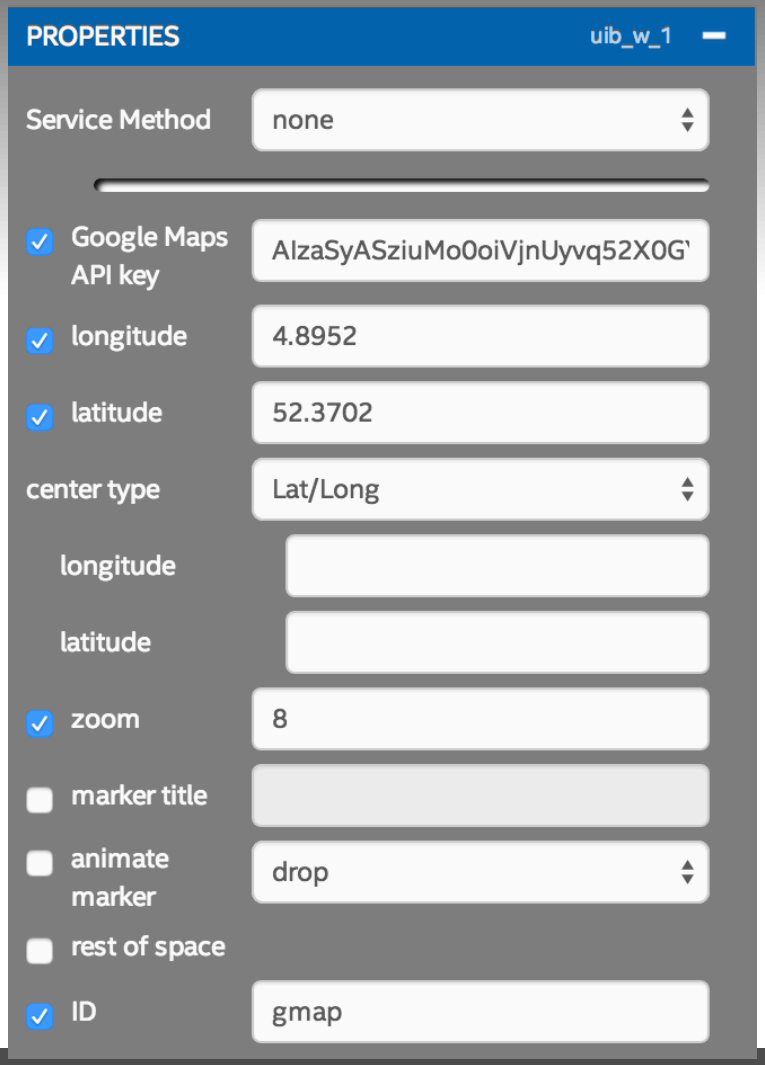
- When user taps on button show user's location on a map
- Tips:
  - Need an API key at: https://developers.google.com/maps/documentation/javascript/
  - Find Google Maps on "Design" & use it
  - Set Google Map properties
  - Don't forget the necessary plugin in your project!



Exercise 2b: Solution

- id="gmap"





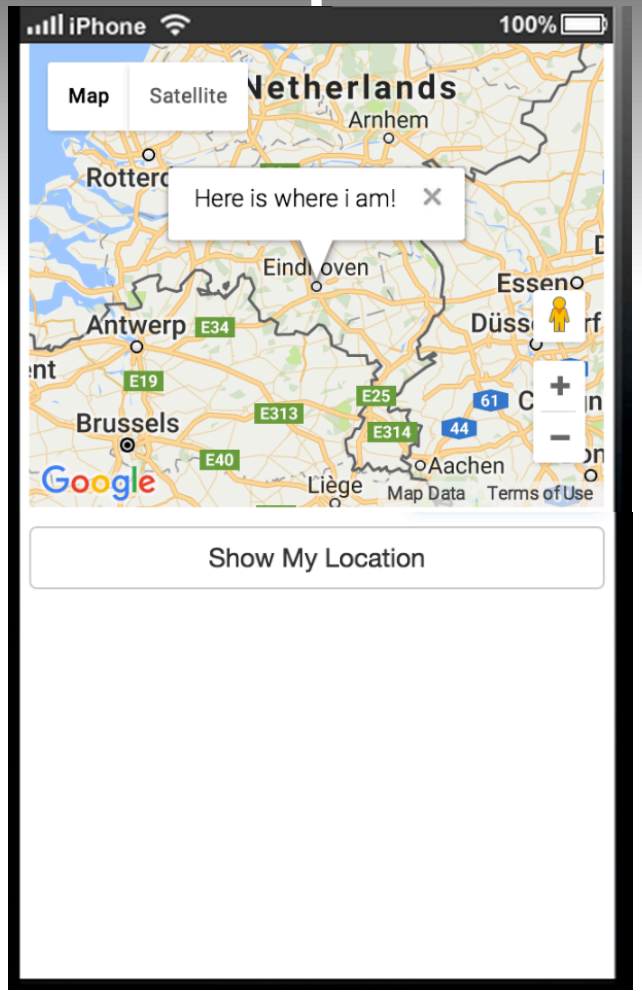
Exercise 2b: Solution

```
11      $(document).on("click", ".uib_w_2", function(evt)
12      {
13          navigator.geolocation.getCurrentPosition(onGeoLocSuccess, onGeoLocError);
14          /* your code goes here */
15           return false;
16      });
17
18       function onGeoLocSuccess(position) {
19           var map = new google.maps.Map(document.getElementById('gmap'), {zoom: 6});
20           var pos = {
21                lat: position.coords.latitude,
22                lng: position.coords.longitude
23           };
24           map.setCenter(pos);
25      }
26      // onGeoLocError Callback receives a PositionError object
27      //
28      function onGeoLocError(error) {
29          alert('code: '    + error.code    + '\n' + 'message: ' + error.message + '\n');
30      }
```


TU/e

Exercise 2c: Location & Maps

- Add an InfoWindow on the map
- Tips:
  - Check out: https://developers.google.com/maps/documentation/javascript/geolocation

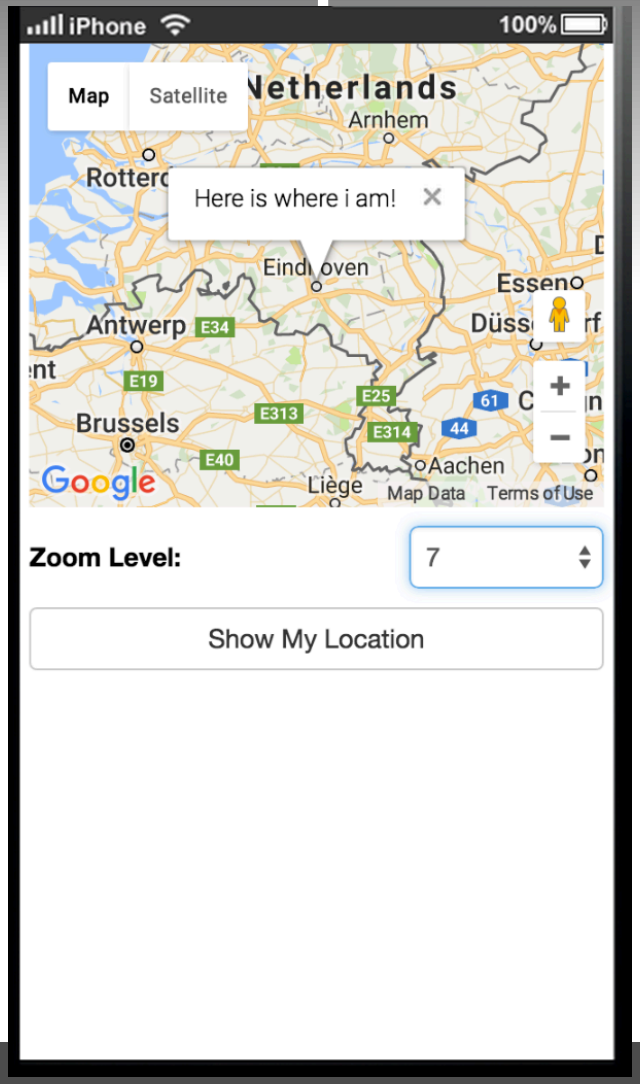


Exercise 2c: Solution

```
var infoWindow = new google.maps.InfoWindow({map: map});
infoWindow.setPosition(pos);
infoWindow.setContent('Here is where i am!');
map.setCenter(pos);
```

Exercise 2d: Location & Maps

- Add custom zoom control on the map
- Tips:
  - Check out: https://api.jquery.com/change/



Exercise 2d: Solution

```
8   function register_event_handlers()
9   {
10
11      var map;
12      var infoWindow;
13      $(document).on("click", ".uib_w_2", function(evt)
14      {
15          map = new google.maps.Map(document.getElementById('gmap'), {zoom: 6});
16          navigator.geolocation.getCurrentPosition(onGeoLocSuccess, onGeoLocError);
17      });
18
19      $('#zoomLevel').change(function(){
20          var userSelected =parseInt($('#zoomLevel option:selected').text());
21          map.setZoom(userSelected);
22      });
23
24      function onGeoLocSuccess(position) {
25           var pos = {
26                lat: position.coords.latitude,
27                lng: position.coords.longitude
28           };
29           infoWindow = new google.maps.InfoWindow({map: map});
30           infoWindow.setPosition(pos);
31           infoWindow.setContent('Here is where i am!');
32           map.setCenter(pos);
33      }
34      // onGeoLocError Callback receives a PositionError object
35      //
36      function onGeoLocError(error) {
37          alert('code: ' + error.code    + '\n' + 'message: ' + error.message + '\n');
38      }
39  }
```

TU/e

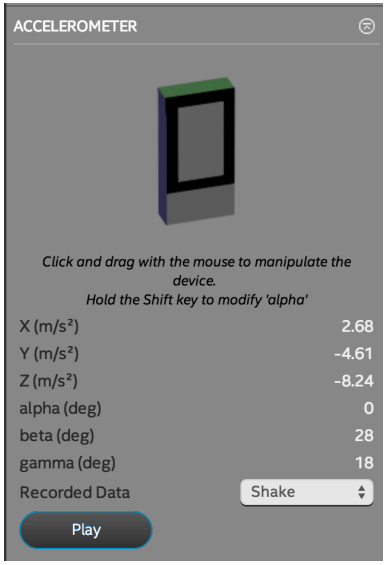
Tips for Google Maps

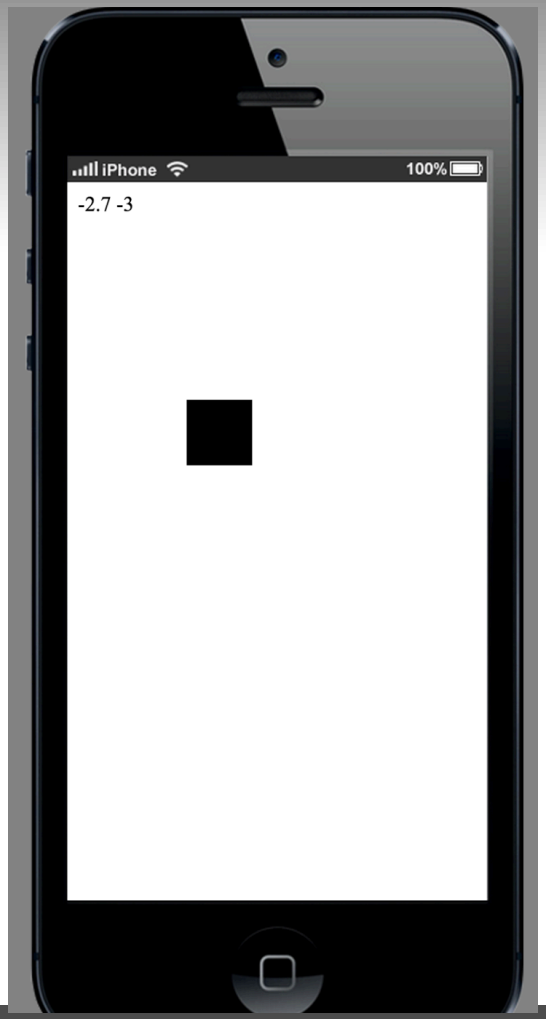
- Hello World:
  - https://developers.google.com/maps/documentation/javascript/tutorial
- Overlays:
  - https://developers.google.com/maps/documentation/javascript/overlays
  - Info Windows:
    - https://developers.google.com/maps/documentation/javascript/infowindows
- Code example: On Wiki
- Video: http://youtu.be/k0cl9oLq-_I

TU/e

Exercise 3: Accelerometer

- When user tilts device, a black square in screen moves accordingly
- Tips:
  - Emulator is handy:
  - Use the project on Wiki to start





Exercise 3: Solution

HTML

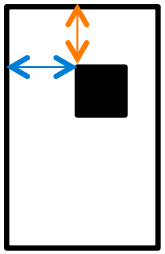
```html
<body>
   <div class="sensorValues"></div>
   <div class="middleBlock"></div>
</body>
```

JAVASCRIPT

```javascript
//get sensor values
var tiltLR = -acceleration.x;
var tiltFB = acceleration.y;

var changeX = screen.width/2-25 + tiltLR*(screen.width/20) + "px";
$('.middleBlock').css('left', changeX);
var changeY = screen.height/2-25 + tiltFB*(screen.height/20) + "px";
$('.middleBlock').css('top', changeY);
```

CSS

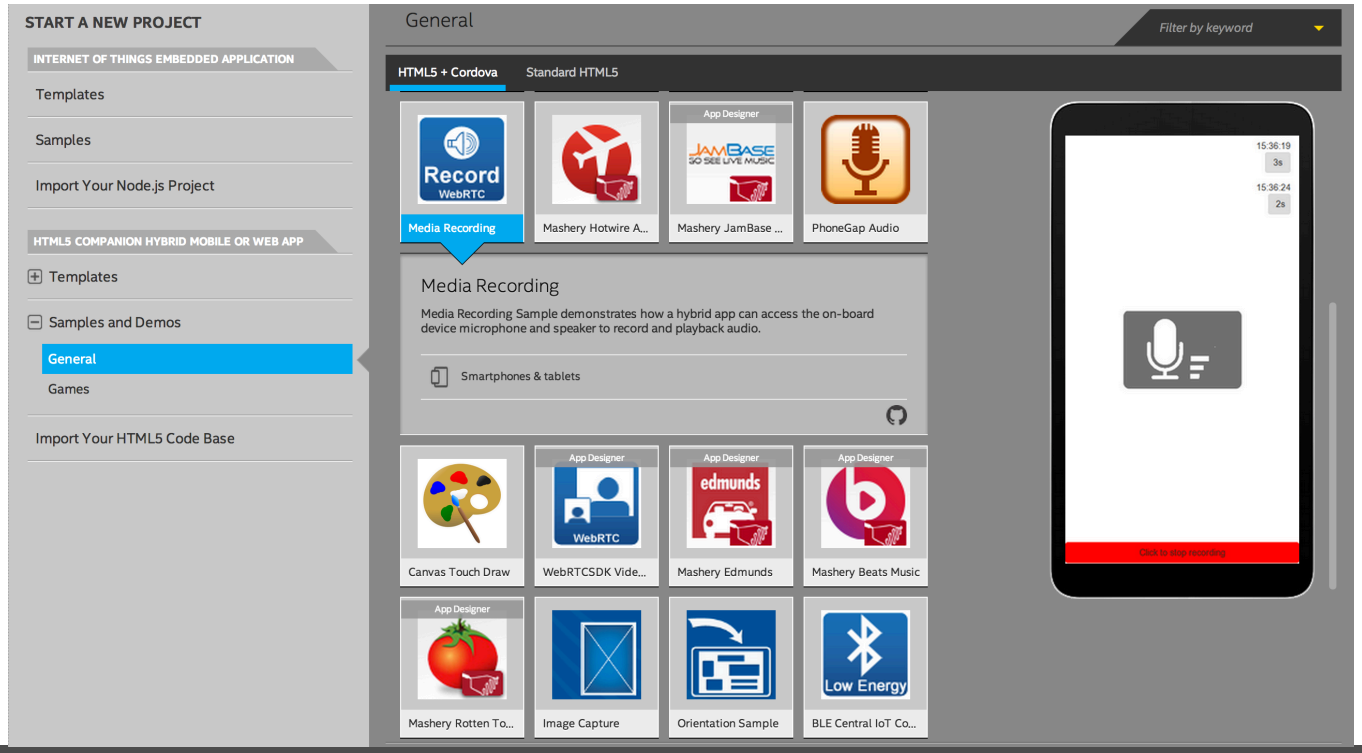
```css
.middleBlock{
  position: absolute;
  width: 50px;
  height: 50px;
  background-color: black;
  left: 0px;
  top: 0px;
}
```



TU/e

Other Sensors?

- Check out “Samples and Demos”
  - INTEL XDK=>”START A NEW PROJECT”=>”Samples and Demos”=>”General”



Summary

- Worked with Sensors
  - Camera
  - Location & Maps
  - Accelerometer