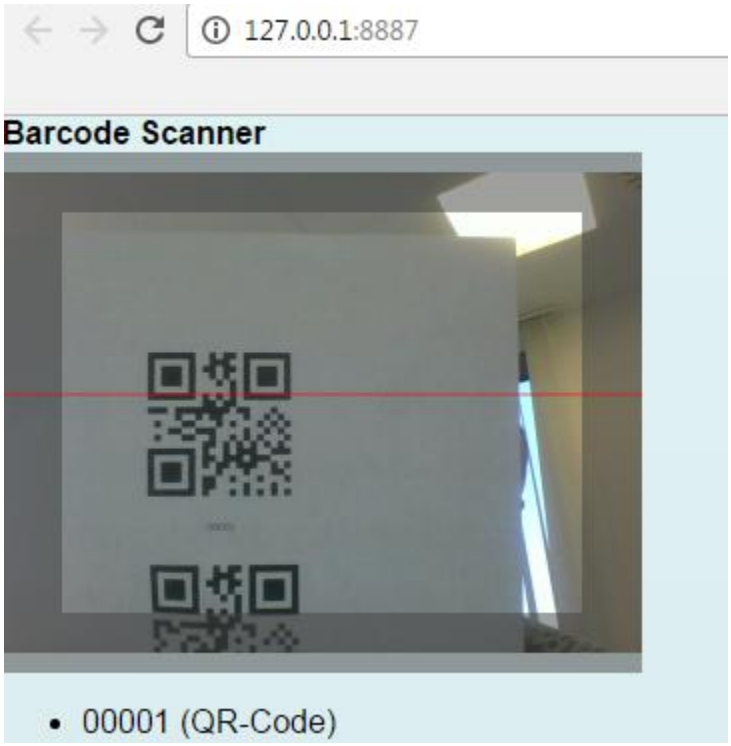
Tworzenie aplikacji progresywnej z UI5

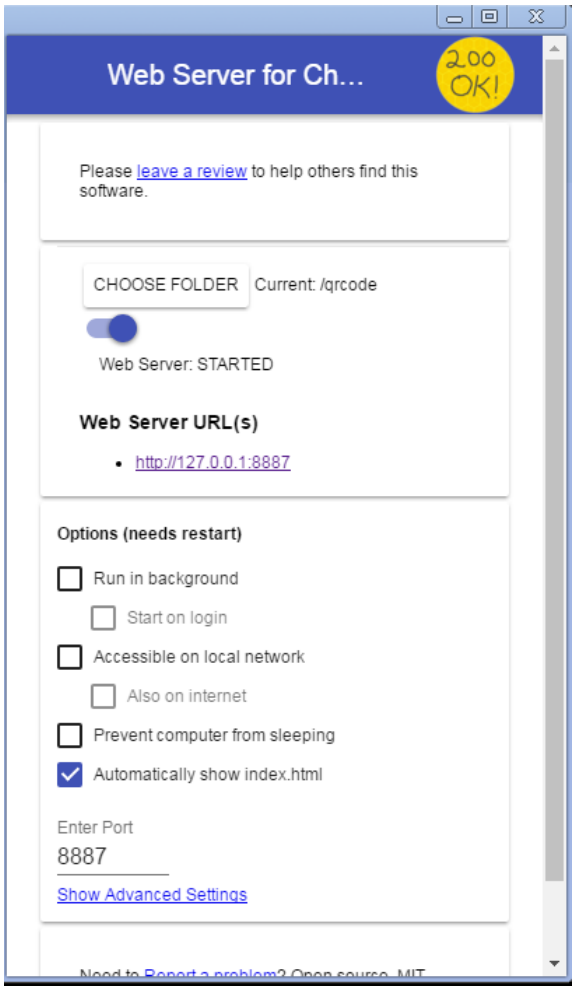
Co chcemy osiągnąć:
Progresywna aplikacja z UX Experience UI5, skanująca kod QR i wyświetlająca poniżej zeskanowaną wartość.



Wymagania:
Zadanie wykonywane będzie w oparciu o Google Chrome.

Przygotowanie środowiska
1. Instalujemy Chrome Web Server
https://chrome.google.com/webstore/detail/web-server-for-chrome/ofhbbkphhbklhfoeikjpcbhemlocgigb?hl=en

Otrzymać powinniśmy następującą aplikację:



2. Zakładamy na naszym dysku, lokalny folder w którym będziemy prowadzić nasz development - np. pwa_ui5.

3. Tworzenie zaczniemy trochę od końca. W celu odczytania słowa zaszytego w kodzie QR posłużymy się otwartą biblioteką, którą znajdziemy pod adresem https://gist.github.com/jazzido/9435670#file-zbar-processor-js. Pobieramy plik `zbar-processor.js` i zapisujemy go w naszym folderze pod tą samą nazwą.

4. Teraz pozostaje nam jedynie stworzyć stronę która uruchomi kamerę i wywołać metodę z biblioteki, która odczyta nam kod QR.

5. W dowolnym edytorze (nawet w notatniku) tworzymy stronę index.html. Nie będę opisywał wszystkich znaczników, bo to jest trywialne, podam jedynie co należy zamieścić, aby uzyskać oczekiwaną funkcjonalność.

6. W headerze wrzucimy bezpośrednio odwołanie do pliku css ze stylami, lub od razu możemy je wrzucić tam. Znacznik ***redline***, odpowiada za czerwoną poprzeczną kreskę na skanerze, znacznik ***inner*** to ciemno szare obramowanie wokół obrazu, ***video*** to ramka z obrazem z kamery.

```
1.    <style type="text/css">
```

```
2.          body > div {
3.            position: relative;
4.            width: 320px; height: 260px;
5.          }
6.
7.          video { position: absolute; top: 0; left: 0; width: 320px; height: 260px; }
8.          div#inner {
9.            position: absolute;
10.           margin: 0 auto;
11.           top: 0; left: 0;
12.           width: 260px; height: 200px;
13.           border: 30px solid rgba(64,64,64, 0.5);
14.           zindex: 1000;
15.         }
16.
17.
18.
19.         div#redline {
20.           position: absolute;
21.           top: 120px;
22.           width: 320px;
23.           height: 2px;
24.           background-color: rgba(255, 0, 0, 0.3);
25.           zindex: 1001;
26.         }
27.  </style>
```

7. Również w headerze, publikujemy odwołanie do bibliotek ui5.

```
1.    <script src="https://openui5.hana.ondemand.com/resources/sap-ui-core.js"
2.            id="sap-ui-bootstrap"
3.            data-sap-ui-theme="sap_bluecrystal"
4.            data-sap-ui-libs="sap.m">
5.    </script>
```

8. Teraz pozostaje nam "jedynie" wyświetlić obraz z kamery i próbować znaleźć kod QR na ekranie.

Funkcja snapshot odpowiedzialna jest za wyłapanie klatki z obrazu i odczytania z niej kodu QR. Metoda `setInterval` odpowiedzialna jest za robienie zrzutu obrazu z kamery co określony czas.

Linie 43-60 odpowiadają za wyświetlenie obrazu na ekranie. API JS `navigator.getUserMedia` pozwala na nawiązanie komunikacji z kamerą i mikrofonem. Aby uzyskać dostęp do urządzenia, należy przekazać parametr `{video: true}`. Pozostały kod odpowiedzialny jest za prezentacje obrazu na ekranie.

```
1.  <body class="sapUiBody" id="content">
2.  <b> Barcode Scanner </b>
3.
4.        <div>
5.        <video autoplay></video>
6.        <div id="inner"></div>
7.        <div id="redline">
8.        </div>
9.        </div>
10.       <ul id="decoded">
11.       </ul>
12.       <canvas style="display:none;"></canvas>
```

```
13.
14.       <script type="text/javascript">
15.         var video = document.querySelector('video');
16.         var canvas = document.querySelector('canvas');
17.         var ctx = canvas.getContext('2d');
18.         var localMediaStream = null;
19.         var list = document.querySelector('ul#decoded');
20.         var worker = new Worker('zbar-processor.js');
21.         worker.onmessage = function(event) {
22.             if (event.data.length == 0) return;
23.             var d = event.data[0];
24.             var entry = document.createElement('li');
25.             entry.appendChild(document.createTextNode(d[2] + ' (' + d[0] + ')'));
26.             list.appendChild(entry);
27.         };
28.
29.         function snapshot() {
30.             if (localMediaStream === null) return;
31.             var k = (320 + 240) / (video.videoWidth + video.videoHeight);
32.             canvas.width = Math.ceil(video.videoWidth * k);
33.             canvas.height = Math.ceil(video.videoHeight * k);
34.             var ctx = canvas.getContext('2d');
35.             ctx.drawImage(video, 0, 0, video.videoWidth, video.videoHeight,
36.                           0, 0, canvas.width, canvas.height);
37.             var data = ctx.getImageData(0, 0, canvas.width, canvas.height);
38.             worker.postMessage(data);
39.         }
40.
41.         setInterval(snapshot, 500);
42.
43.         navigator.getUserMedia = navigator.getUserMedia || navigator.webkitGetUserMedia ||
    navigator.mozGetUserMedia || navigator.msGetUserMedia;
44.         window.URL = window.URL || window.webkitURL || window.mozURL || window.msURL;
45.         if (navigator.getUserMedia) {
46.             navigator.getUserMedia({video: true},
47.                                    function(stream) { // success callback
48.                                        if (video.mozSrcObject !== undefined) {
49.                                            video.mozSrcObject = stream;
50.                                        } else {
51.                                            video.src = (window.URL &&
    window.URL.createObjectURL(stream)) || stream;
52.                                        }
53.                                        localMediaStream = true;
54.                                    },
55.                                    function(error) {
56.                                        console.error(error);
57.                                    });
58.         }
59.         else {
60.         }
61.         </script>
62.
63. </body>
```

9. Aplikacja jest gotowa, teraz wystarczy jedynie ją uruchomić. Otwórz Web Server for Chrome. Status serwera powinien być ustawiony na "Started". Następnie wystarczy uruchomić aplikację z zaznaczonego w WSfC linka.


Web Server for Ch...
200 OK!
Please leave a review to help others find this software.
CHOOSE FOLDER
Current: /qrcode
Web Server: STARTED
Web Server URL(s)
http://127.0.0.1:8887