<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml"

xmlns:v="urn:schemas-microsoft-com:vml">

<head>

<meta name="viewport" content="initial-scale=1.0, user-scalable=no" />

<style type="text/css">

html {

height: 100%

}

body {

height: 100%;

margin: 0;

padding: 0

}

#pano\_canvas {

height: 100%

}

</style>

<script type="text/javascript"

src="http://maps.googleapis.com/maps/api/js?key=AIzaSyDCa5eo1WwwMqtRkAE38e\_q\_qWix83plUw&sensor=false">

</script>

<script type="text/javascript">

// Views

var map;

var panorama;

var client;

var entryPanoId = null;

// Path

var polyLine;

// Positions

var entryPos = new google.maps.LatLng(42.077529, -88.223569);

var lobbyPos = new google.maps.LatLng(42.077457, -88.214979);

var atriumPos = new google.maps.LatLng(42.077410, -88.214142);

var loungePos = new google.maps.LatLng(42.076693, -88.213992);

var cubePos = new google.maps.LatLng(42.076693, -88.215408);

var officePos = new google.maps.LatLng(42.076661, -88.214614);

// JSON database

var database =

{

"position":

{

"entry": new google.maps.LatLng(42.077529, -88.223569),

"lobby": new google.maps.LatLng(42.077457, -88.214979),

"atrium": new google.maps.LatLng(42.077410, -88.214142),

"lounge": new google.maps.LatLng(42.076693, -88.213992),

"office": new google.maps.LatLng(42.076661, -88.214614),

"cube": new google.maps.LatLng(42.076693, -88.215408)

},

"mapOptions":

{

"center": entryPos,

"zoom": 17,

"maxzoom": 19,

"mapTypeControl": false,

"zoomControl": false,

"mapTypeId": google.maps.MapTypeId.HYBRID,

"streetViewControl": true,

"disableDefaultUI": true

},

"panoOptions":

{

"position": entryPos,

"visible": true,

"disableDoubleClickZoom": false,

"zoomControl": true,

"zoomControlOptions": {

style: google.maps.ZoomControlStyle.LARGE,

position: google.maps.ControlPosition.LEFT\_CENTER

},

//"panoProvider": getCustomPanorama,

"panoProvider": getAutoPanorama,

"pov": {

heading: 0,

pitch: 0,

zoom: 0

}

},

"polyLineOptions":

{

"map": map,

"strokeColor": "red",

"strokeWeight": 5,

"strokeOpacity": 0.5,

"visible": true,

"path": [lobbyPos, atriumPos, loungePos, officePos, cubePos]

},

"markerData": [

{"title": "entry", "position": entryPos},

{"title": "lobby", "position": lobbyPos},

{"title": "atrium", "position": atriumPos},

{"title": "lounge", "position": loungePos},

{"title": "office", "position": officePos},

{"title": "cube", "position": cubePos}

],

"images":

{

//"redIcon": "http://skin-dev.searshc.com/Skin20ISV/images/redIcon.png",

//"greenIcon": "http://skin-dev.searshc.com/Skin20ISV/images/greenIcon.png",

//"blueIcon": "http://skin-dev.searshc.com/Skin20ISV/images/blueIcon.png"

"redIcon": "images/redIcon.png",

"greenIcon": "images/greenIcon.png",

"blueIcon": "images/blueIcon.png"

},

"customPano":

{

"entry":

{

"pano": null,

"links":

[

{

'heading': 70,

'description' : 'Go To Lobby',

'pano' : 'lobby',

}

]

},

"lobby":

{

"pano": 'lobby',

"description": "SHC Lobby",

"latLng": new google.maps.LatLng(42.077457, -88.214979),

"links":

[

{

'heading': 0,

'description' : "Atrium",

'pano' : 'atrium'

},

{

'heading': 180,

'description' : 'Exit',

'pano' : entryPanoId

}

],

"copyright": "(c) 2012 SHC",

"tileSize": new google.maps.Size(256, 256),

"worldSize": new google.maps.Size(4096, 2048),

"centerHeading": 0,

"getTileUrl": getCustomPanoramaTileUrl

},

"atrium":

{

"pano": 'atrium',

"description": "SHC Atrium",

"latLng": new google.maps.LatLng(42.077410, -88.214142),

"links":

[

{'heading': 45, 'description' : 'Lounge', 'pano' : 'lounge'},

{'heading': 180, 'description' : 'Lobby', 'pano' : 'lobby'}

],

"copyright": "(c) 2012 SHC",

"tileSize": new google.maps.Size(256, 256),

"worldSize": new google.maps.Size(4096, 2048),

"centerHeading": 0,

"getTileUrl": getCustomPanoramaTileUrl

},

"lounge":

{

"pano": 'lounge',

"description": "E5 Lounge",

"latLng": new google.maps.LatLng(42.076693, -88.213992),

"links":

[

{'heading': 90, 'description' : 'Office', 'pano' : 'office'},

{'heading': 180, 'description' : 'Atrium', 'pano' : 'atrium'}

],

"copyright": "(c) 2012 SHC",

"tileSize": new google.maps.Size(256, 256),

"worldSize": new google.maps.Size(4096, 2048),

"centerHeading": 0

},

"office":

{

"pano": 'office',

"description": "E5 Office",

"latLng": new google.maps.LatLng(42.076661, -88.214614),

"links":

[

{'heading': 0, 'description' : 'Cube', 'pano' : 'cube'},

{'heading': 180, 'description' : 'Lounge', 'pano' : 'lounge'}

],

"copyright": "(c) 2012 SHC",

"tileSize": new google.maps.Size(256, 256),

"worldSize": new google.maps.Size(4096, 2048),

"centerHeading": 0

},

"cube":

{

"pano": 'cube',

"description": "E5 Cube",

"latLng": new google.maps.LatLng(42.076693, -88.215408),

"links": [{'heading': 180, 'description' : 'Office', 'pano' : 'office'}],

"copyright": "(c) 2012 SHC",

"tileSize": new google.maps.Size(256, 256),

"worldSize": new google.maps.Size(4096, 2048),

"centerHeading": 180

}

}

}

// Store Marker Objects

var markers = [];

var pin = {};

// Sets up the map, panorama, path line, entry point, and markers

function initialize()

{

setUpMapCanvas();

setUpPanoCanvas();

setUpPolyLinePath();

setUpEntryPano();

setUpMarkers();

setBlueIcon(pin.entry);

}

// The google maps entry point is verified and its panoId is stored in entryPanoId

function setUpEntryPano()

{

var panoSearchRadius = 50;

client = new google.maps.StreetViewService();

client.getPanoramaByLocation(database.position.entry, panoSearchRadius, function(result, status) {

if (status == google.maps.StreetViewStatus.OK) {

entryPanoId = result.location.pano;

database.customPano.entry.pano = entryPanoId;

}

});

}

// Create the google map

function setUpMapCanvas()

{

var mapOptions =

{

center: database.mapOptions.center,

zoom: database.mapOptions.zoom,

maxZoom: database.mapOptions.maxZoom,

mapTypeControl: database.mapOptions.mapTypeControl,

zoomControl: database.mapOptions.zoomControl,

mapTypeId: database.mapOptions.mapTypeId,

streetViewControl: database.mapOptions.streetViewControl,

disableDefaultUI: database.mapOptions.disableDefaultUI

};

map = new google.maps.Map(document.getElementById('map\_canvas'), mapOptions);

}

// Create the panorama

function setUpPanoCanvas()

{

var panoOptions =

{

position: database.panoOptions.position,

visible: database.panoOptions.visible,

disableDoubleClickZoom: database.panoOptions.disableDoubleClickZoom,

zoomControl: database.panoOptions.zoomControl,

zoomControlOptions: database.panoOptions.zoomControlOptions,

panoProvider: database.panoOptions.panoProvider,

pov: database.panoOptions.pov

}

panorama = new google.maps.StreetViewPanorama(document.getElementById("pano\_canvas"), panoOptions);

google.maps.event.addListener(panorama, 'links\_changed', createCustomLinks);

}

// Create the path between positions

function setUpPolyLinePath()

{

var polyLineOptions =

{

map: map,

strokeColor: database.polyLineOptions.strokeColor,

strokeOpacity: database.polyLineOptions.strokeOpacity,

strokeWeight: database.polyLineOptions.strokeWeight,

path: database.polyLineOptions.path

}

polyLine = new google.maps.Polyline(polyLineOptions);

}

// Create markers for each position

function setUpMarkers()

{

map.setStreetView(panorama);

for (index in database.markerData)

{

var markerOptions =

{

title: database.markerData[index].title,

position: database.markerData[index].position,

map: map,

icon: database.images.redIcon

};

var marker = new google.maps.Marker(markerOptions);

markers.push(marker);

pin[marker.title] = marker;

setUpListener(marker);

}

}

// Create listener on marker to change panorama on click

function setUpListener(marker)

{

google.maps.event.addListener(marker, 'click', function(){

var peg;

var pano;

if(marker.title == 'entry')

{

pano = entryPanoId;

}

else

{

pano = marker.title

}

panorama.setPano(pano)

panorama.setPov({heading: 0, pitch: 0, zoom:0});

setBlueIcon(marker);

map.setStreeView(pano);

})

}

// Set the marker blue to indicate current location

function setBlueIcon(marker)

{

for (index in markers)

{

if(markers[index].title != marker.title)

{

//markers[index].setVisible(true);

markers[index].setIcon(database.images.redIcon);

markers[index].setAnimation(google.maps.Animation.null);

}

else

{

//markers[index].setVisible(false);

markers[index].setIcon(database.images.blueIcon);

markers[index].setAnimation(google.maps.Animation.BOUNCE);

}

}

}

// Retrieve panorama tiles based on zoom level and orientation

function getCustomPanoramaTileUrl(pano,zoom,tileX,tileY)

{

var url = 'http://skin-dev.searshc.com/Skin20ISV/images/'+pano+'-tiles'+'/'+pano+'\_'+zoom+'\_'+tileX+'\_'+tileY+'.jpg';

var dir = 'images/'+pano+'-tiles'+'/'+pano+'\_'+zoom+'\_'+tileX+'\_'+tileY+'.jpg';

return dir;

}

// Automatically retrieve custom panorama information

function getAutoPanorama(pano)

{

//alert("2:"+database.customPano[pano].pano);

if(pano == entryPanoId)

{

//map.panTo(panorama.getPosition());

setBlueIcon(pin.entry);

return null;

}

else if (pano in database.customPano)

//else if (database.customPano[pano].pano == pano)

{

var customPano = database.customPano[pano];

map.panTo(database.position[pano]);

setBlueIcon(pin[pano]);

return {

location:

{

pano: pano,

description: customPano.description,

latLng: customPano.latLng

},

links: customPano.links,

copyright: customPano.copyright,

tiles:

{

tileSize: customPano.tileSize,

worldSize: customPano.worldSize,

centerHeading: customPano.centerHeading,

getTileUrl: getCustomPanoramaTileUrl

}

};

}

else

{

map.panTo(panorama.getPosition());

pin.entry.setIcon(database.images.redIcon);

pin.entry.setAnimation(google.maps.Animation.null);

return null;

}

}

function createCustomLinks()

{

if (entryPanoId)

{

var links = panorama.getLinks();

var panoId = panorama.getPano();

switch(panoId)

{

case entryPanoId:

map.panTo(entryPos);

links.push(database.customPano.entry.links[0])

break;

case 'lobby':

map.panTo(lobbyPos);

links.push

({

'heading': 0,

'description' : 'Atrium',

'pano' : 'atrium'

});

links.push({

'heading': 180,

'description' : 'Exit',

'pano' : entryPanoId

});

break;

default:

return;

}

}

}

// Toggle the google map from view

function toggleMapView()

{

if(document.getElementById('map\_canvas').style.visibility == 'hidden')

{document.getElementById('map\_canvas').style.visibility = 'visible';}

else

{document.getElementById('map\_canvas').style.visibility = 'hidden';}

}

google.maps.event.addDomListener(window, 'load', initialize);

</script>

</head>

<body onload="initialize()">

<div id="pano\_canvas"

style="z-index: 0; position: absolute; left: 0px; top: 0px; width: 100%; height: 100%">

</div>

<div id="map\_canvas"

style="z-index: 1; position: absolute; right: 0px; bottom: 0px; width: 200px; height: 200px"></div>

<div id="toggle">

<input style="z-index: 2; position: absolute; left: 0px; bottom: 0px"

type="button" value="Toggle Map View" onclick="toggleMapView();"></input>

</div>

<!--

<div id="pano\_canvas" style="z-index:0; width: 100%; height: 100%"></div>

<div id="map\_canvas" style="z-index:1; position:absolute; left:0px; bottom:0px; width: 100%; height: 100%"></div>

-->

</body>

</html>