

**“Ģeoportāla (geolatvija.lv) iegultās kartes API trešo pušu  
izstrādātājiem”**

# **Valsts digitālās attīstības aģentūra**



**Valsts digitālās  
attīstības aģentūra**

2024

## 1. Ievads

Ģeoportāls nodrošina iegultās kartes veidošanu izvietošanai trešās puses mājas lapās. Ģeoportālā veidojot iegulto karti, iespējams to pielāgot konkrētam risinājumam, piemēram, nomainīt fona karti, pievienot virslāņus, pievienot noteiktu ģeoprojektu vai TAPIS karti, iecentrēt un pietuvināt karti konkrētam apgabalam, kā arī pievienot marķierus, uz kuriem lietotājs var klikšķināt uz iegūt papildus informāciju iznirstošajā logā.

Tomēr var rasties gadījumi, kad trešās puses mājas lapā kartei ir nepieciešama papildus funkcionalitāte, piemēram, dinamiski jāpievieno marķieri vai jātuvinā karti noteiktai adresei. Šādos gadījumos trešās puses mājas lapas izstrādātāji var izmantot ģeoportāla iegultās kartes programmēšanas saskarni jeb API.

Šajā dokumentā ir aprakstīta API pieejamā funkcionalitāte un instrukcijas kā to pieslēgt.

## 2. Izmantošanas priekšnoteikumi un ierobežojumi

0. Lai izmantotu iegultās kartes API, vispirms ģeoportālā ir jāizveido pati iegultā karte un jāiegūst *iframe* html fragments, kas būs jāizvieto trešās puses mājas lapā. Iegulto karti var izveidot tikai autentificēts lietotājs.

Lietotājs var atvērt iegultās kartes veidošanas sadaļu nospiežot pie rīkiem uz pogas 

Lietotājiem tiks prasīts aizpildīt šādu informāciju:

*Nosaukums – lietotājam ir jāievada iegultās kartes nosaukums*

*Domēns – lietotājam ir jāievada domēns, kurā ir plānots izmantot izveidoto iegultās kartes logu. Lietotājs var ievadīt vairākus domēnus, bet tad tie ir jāatdala ar semikolu (;).*

*Izmērs – lietotājam ir iespējams izvēlēties starp trīs dažādiem standarta izmēriem vai izvēlēties citu izmēru:*

- a. Mazs (300px x 300px)*
- b. Vidējs (425px x 300px)*
- c. Liels (640px x 480px)*
- d. Cits izmērs – lietotājs ievada vēlamās vērtības.*

*Poga “Pievienot marķieri” – lietotājam ir iespēja kartē pievienot atzīmes. Atzīme novietojas kartes centrā, kad tiek nospiesta poga.*

*Poga “Priekšskatīt” – tiek uzģenerēts kods un saglabāts. Jaunā cilnē atveras iegultais kartes logs.*

*Poga “Izveidot kodu” – tiek uzģenerēts kods un saglabāts.*

*Teksta lauks – parāda uzģenerētā HTML kodu*

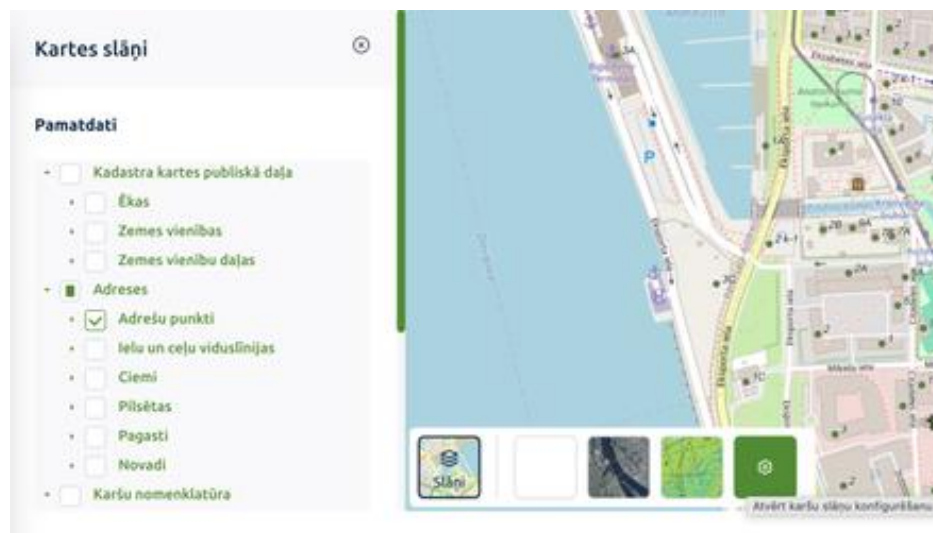
1. Trešās puses mājas lapa ar iegultās kartes logu (*iframe*) sazināsies, izmantojot *postMessage* tehnoloģiju:

<https://developer.mozilla.org/en-US/docs/Web/API/Window/postMessage>

kam ir noteikti pārlūka drošības nosacījumi, piemēram, šajā dokumentā minētie html fragmenti ir jāizvieto uz web servera, lai tos testētu, jo, atverot html lapas pa tiešo pārlūkā no citā diskā, *postMessage* nevarēs pilnvērtīgi izmantot un tiks atgrieztas kļūdas. Var izmantot tiešsaistē izmantojamas izstrādes vietnes, piemēram,

<https://jsfiddle.net/lindsejs/wfouj4nc/45/>

2. Noteikta API funkcionalitāte (“EnableAddressPopups”, “onAddressClick”) ir pieejama tikai tad, kad iegultajā kartē ir redzams adrešu slānis. TAPIS kartē tas nekad nav redzams. Ģeoprojektu kartē to var pieslēgt vai nu iegultās kartes sagatavošanas laikā vai arī pats gala lietotājs. Slānis ir pieslēdzams “Slāņu konfigurēšanas” sadaļā (“Slāņi”->slāņu konfigurēšana (zobrata ikona)-> “Adrešu punkti”). Slānis nav redzams visos mērogos, karte jāprietuvina, lai adrešu punkti tiktu attēloti.



### 3. API pieslēgšana

Lai pieslēgtu ģeoportāla iegultās kartes API:

1. Jāizveido ģeoportālā iegultā karte un jāiegūst *iframe* fragments, kas jāizvieto trešās puses mājas lapas *html* koda *body* bloka vēlamajā vietā, piemēram:

```
<body>
  <iframe
    width="640"
    scrolling="no"
    height="480"
    style="border: 0; overflow: hidden"
    src="https://geolatvija.lv/map/?id=<iegultās kartes identifikators tiek izveidots
kartes veidošanas laikā>"
  ></iframe>
</body>
```

*width* un *height* ir iegultās kartes loga izmēri. Tos var pamainīt pēc vajadzības, piemēram, ja *iframe* bloks ir izvietots kādā noteikta izmēra *html* blokā (*div*), tad *iframe* izmērus var norādīt "100%", lai karte aizņem visu virsobjektu. *src* - jānokopē no iegultās kartes koda fragmenta, kas tiek attēlotas kartes veidošanas laikā.

2. Jāpievieno ģeoportāla *JavaScript* API fails - trešās puses mājas lapas *html* kodā (tipiska *head* blokā) iekļaujot *script* norādi, piemēram:

```
<head>
  <script src="https://geolatvija.lv/map/MapApi"></script>
</head>
```

Svarīgi iekļaut Javascript API no tās pašas ģeoportāla vides, kur tika izveidota iegultā karte, tas ir: produkcijas videi - <https://geolatvija.lv/map/MapApi> testa videi - <https://geolatvija-test.vraa.gov.lv/map/MapApi>

3. *iframe* blokas jāpiešķir unikāls html objekta identifikators (*id*=<unikāla vērtība html lapas ietvaros>), piemēram:

VDA.A.GEO.API.5.0\_EMBED

```
<iframe
  width="640"
  scrolling="no"
  height="480"
  style="border: 0; overflow: hidden"
  src="https://geolatvija.lv/map/?id=<iegultās kartes identifikators tiek izveidots
kartes veidošanas laikā>"
  id="geolatvija-map-modal"
></iframe>
```

4. jāizveido *Javascript* funkcija, kurā būs pieejams “map” objekts API funkcionalitātes lietošanai:

```
<script>
function initGeolatvija(map) {
  // API lietošana, piemēram:
  map.ZoomToPoint(510982.62, 311289.86);
}
</script>
```

5. un jāinicializē API no <https://geolatvija.lv/map/MapApi> , norādot iframe identifikatoru, piemēram:

```
<script>
function initGeolatvija(map) {
  // API lietošana, piemēram:
  map.ZoomToPoint(510982.62, 311289.86);
}
mapAPI("geolatvija-map-modal", initGeolatvija);
</script>
```

### 3.1. API lietošanas pilns paraugs

Pilns API lietošanas paraugs (atkomentēta ir kartes iecentrēšana uz Daugavpils Centrālo parku):

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <meta name="theme-color" content="#000000" />
    <script src="https://geolatvija.lv/map/MapApi"></script>
    <title>GEO</title>
  </head>
  <body>
    <noscript>You need to enable JavaScript to run this app.</noscript>
    <iframe
      width="640"
      scrolling="no"
      height="480"
      style="border: 0; overflow: hidden"
      src="https://geolatvija.lv/map/?id=d83d615a-8f57-4865-9817-a95098ad18d7"
      id="geolatvija-map-modal"
    ></iframe>
  </body>
  <script>
function initGeolatvija(map) {// iecentrē karti uz Daugavpils Centrālo parku:
  map.ZoomToPoint(658178.18, 194527.16);

  // iecentrē un pietuvina karti administratīvajai teritorijai, Gulbenei:
  // map.ZoomToAddress(["0029200"], []);
  // tā lai kartē ir redzama gan Gulbene, gan Alūksne:
  // map.ZoomToAddress(["0029200", "0021200"], []);
  // pietuvina un iecentrē adresi pēc VZD adreses koda:
  // map.ZoomToAddress([], ["106570942"]);
  // pietuvina un iecentrē gan adresi, gan admin. teritoriju (Tukums un adrese Rīgā)
  // map.ZoomToAddress(["0052210"], ["106570942"]);

  // izveido marķierus
  /*
  var markers = [
    { x: 506646, y: 312794, description: "Alberta iela 10, Rīga" },
```

```

{
  x: 506686,
  y: 312764,
  description: '<div class="navlink"><a href="https://www.google.com/maps/search/?api=1&query="
target="_blank">How to get</a></div>',
  color: "rgba(170, 30, 200, 1.0)",
},
];
// attēlo marķierus uz kartes
map.ShowMarkers(markers);
// iecentrē karti koordinātām, kur ir marķieri
map.ZoomToPoint(506646, 312794);
// atspējo/nerāda marķieru iznirstošos logus
//map.EnableMarkerPopups(false);
map.RegisterForEvent("onMarkerClick", function (x, y, markers) {
  console.log("onMarkerClick");
  console.log({ x, y, markers });
  //map.UnRegisterForEvent("onMarkerClick");
});
*/

// iespējo adresu iznirstošos logus
//map.EnableAddressPopups(true);

// atspējo ģeoportāla marķieru iznirstošos logus
// map.EnableApplicationMarkerPopups(false);

/*
map.RegisterForEvent("onMapClick", function (longitude, latitude) {
  console.log("onMapClick");
  console.log({ longitude: longitude, latitude: latitude });
  alert("Koordināta: "+ longitude + " " + latitude);
  //console.log("UnRegisterForEvent for onMapClick");
  //map.UnRegisterForEvent("onMapClick");
});
*/
/*

```



```

// klausās ģeoportālā iegultās kartes veidošanas laikā izvietoto marķieru klikšķus
map.RegisterForEvent(
  "onApplicationMarkerClick",
  function (x, y, markers) {
    console.log("onApplicationMarkerClick");
    console.log({ x, y, markers });
    //map.UnRegisterForEvent("onApplicationMarkerClick");
  }
);
*/
/*
// klausās adresu punktu klikšķus
map.RegisterForEvent("onAddressClick", function (x, y, addresses) {
  console.log("onAddressClick");
  console.log({ x, y, addresses });
  alert("Adrese: "+ addresses[0].std);
  //map.UnRegisterForEvent("onAddressClick");
});
*/
}
mapAPI("geolatvija-map-modal", initGeolatvija);
</script>
</html>

```

### 3.2. Pieejamā API funkcionalitāte

“mapApi” inicializēšanas funkcijas atgrieztais kartes objekts (piemēros “map”) nodrošina šādas funkcijas:

- tūlītējas darbības ar karti - **“ZoomToPoint”, “ZoomToAddress”, “ShowMarkers”**;
- kartes iestatījumu maiņu - **“EnableAddressPopups”, “EnableMarkerPopups”, “EnableApplicationMarkerPopups”**;

- kartes notikumi - uzsākt un beigt klausīties: **“RegisterForEvent“**, **“UnRegisterForEvent“**, notikumus **“onMapClick“**, **“onMarkerClick“**, **“onApplicationMarkerClick“**, **“onAddressClick“**.

## 4. Darbības ar karti

### 4.1. Tūlītējas darbības ar karti

#### 4.1.1 ZoomToPoint

Funkcijai jāpārdod koordinātas LKS-92 jeb EPSG:3059 koordinātu sistēmā (secībā - garums, platums). Rezultātā karte tiek iecentrēta uz norādītajām koordinātām nemainot kartes pietuvināšanas soli (tas ir, mērogs paliek tāds, kā tika iestatīts veidojot iegulto karti geolattvija.lv).

Piemērs iecentrē karti uz Daugavpils Centrālo parku:

```
<script>
function initGeolattvija(map) {
    // iecentrē karti uz Daugavpils Centrālo parku:
    map.ZoomToPoint(658178.18, 194527.16);
}
mapAPI("geolattvija-map-modal", initGeolattvija);
</script>
```

#### 4.1.2 ZoomToAddress

Funkcijai jāpārdod divi masīvi:

- pirmajā var norādīt administratīvo teritoriju ATVK kodus, uz kurām pietuvināt/iecentrēt karti, ATVK saraksts: <https://www.csp.gov.lv/lv/klasifikacija/atvk-2021/administrativo-teritoriju-un-teritoriala-iedalijuma-vienibu-klasifikators>
- otrajā var norādīt adresu VZD kodus, uz kurām pietuvināt/iecentrēt karti:

Piemērs karti iecentrē/pietuvina tā, lai redzams Tukums (ATVK 0052210 ) un adrese Rīgā (ar VZD kodu 106570942):

```
<script>
```

VDAA.GEO.API.5.0\_EMBED

```

function initGeolatvija(map) {
    // iecentrē un pietuvina karti administratīvajai teritorijai, Gulbenei:
    // map.ZoomToAddress(["0029200"], []);
    // tā lai kartē ir redzama gan Gulbene, gan Alūksne:
    // map.ZoomToAddress(["0029200", "0021200"], []);
    // pietuvina un iecentrē adresi pēc VZD adreses koda:
    // map.ZoomToAddress([], ["106570942"]);
    // pietuvina un iecentrē gan adresi, gan admin. teritoriju (Tukums un adrese Rīgā)
    map.ZoomToAddress(["0052210"], ["106570942"]);
}
mapAPI("geolatvija-map-modal", initGeolatvija);
</script>

```

### 4.1.3 ShowMarkers

Funkcijai jāpadod masīvs ar marķieriem (atsaucēm), kuri tiks attēloti kartē.

Marķieris ir *JavaScript* objekts ar šādiem laukiem:

- x - LKS-92 jeb EPSG:3059 koordinātu sistēmā garums (obligāts);
- y - LKS-92 jeb EPSG:3059 koordinātu sistēmā platums (obligāts);
- description - marķiera apraksts, kas tiks attēlots iznirstošajā logā, kad lietototājs to kartē izvēlēsies klikšķinot (neobligāts);  
var saturēt *html* kodu;  
ja "description" netiek norādīts, tad iznirstošais logs netiks attēlots;
- color - marķiera ikonas krāsa (neobligāts);

Piemērs, kas kartē novieto divus marķierus ar aprakstiem un iecentrē karti uz marķiera koordinātām:

```

<script>
function initGeolatvija(map) {
    // izveido marķierus
    var markers = [

```

```
{ x: 506646, y: 312794, description: "Alberta iela 10, Rīga" },  
{  
  x: 506686,  
  y: 312764,  
  description: '<div class="navlink"><a href="https://www.google.com/maps/search/?api=1&query="'  
target="_blank">How to get</a></div>',  
  color: "rgba(170, 30, 200, 1.0)",  
},  
];  
// attēlo marķierus uz kartes  
map.ShowMarkers(markers);  
// iecentrē karti koordinātām, kur ir marķieri  
map.ZoomToPoint(506646, 312794);  
}  
mapAPI("geolatvija-map-modal", initGeolatvija);  
</script>
```

## 4.2. Kartes iestatījumu maiņa

### 4.2.1 EnableAddressPopups

Ja "EnableAddressPopups" tiek iespējots (tas ir, izsaukts ar "true" parametru) un kartē ir pievienots "Adrešu punktu" slānis (skat. "Izmantošanas priekšnoteikumi un ierobežojumi" sadaļu), tad lietotājam klikšķinot uz adreses punkta tiks attēlots iznirstošais logs ar adreses nosaukumu:



Noklusēti adresu iznirstošie logi ir **izslēgti**.

Piemērs, kas iespējo adresu logus:

```
<script>
function initGeolatvija(map) {
    // iespējo adresu iznirstošos logus
    map.EnableAddressPopups(true);
}
mapAPI("geolatvija-map-modal", initGeolatvija);
</script>
```

#### 4.2.2 EnableMarkerPopups

Ļauj iespējot/atspējot ar API izvietoto marķieru (ar "ShowMarkers" funkciju) iznirstošos logus, kad lietotājs klikšķina uz marķiera.

Noklusēti API marķieru iznirstošie logi ir **ieslēgti**.

Logs netiek rādīts, ja marķiera "description" lauks netika aizpildīts.

Piemērs, kas atspējo (lietotājam nerādīs) marķieru logus:

VDA.A.GEO.API.5.0\_EMBED

```

<script>
function initGeolatvija(map) {
  // izveido marķierus
  var markers = [
    { x: 506646, y: 312794, description: "Alberta iela 10, Rīga" },
    {
      x: 506686,
      y: 312764,
      description: '<div class="navlink"><a href="https://www.google.com/maps/search/?api=1&query="
target="_blank">How to get</a></div>',
      color: "rgba(170, 30, 200, 1.0)",
    },
  ];
  // attēlo marķierus uz kartes
  map.ShowMarkers(markers);
  // iecentrē karti koordinātām, kur ir marķieri
  map.ZoomToPoint(506646, 312794);
  // atspējo/nerāda marķieru iznirstošos logus
  map.EnableMarkerPopups(false);
}
mapAPI("geolatvija-map-modal", initGeolatvija);
</script>

```

### 4.2.3 EnableApplicationMarkerPopups

Ļauj iespējot/atspējot to marķieru iznirstošos logus, kas tika pievienoti iegultās kartes izveidošanas laikā ģeoportālā.

Noklusēti ģeoportāla marķieru iznirstošie logi ir **ieslēgti**.

Logs netiek rādīts, ja marķiera apraksts netika aizpildīts, veidojot iegulto karti.

Piemērs, kas atspējo (lietotājam nerādīs) ģeoportāla marķieru logus:

```

<script>
function initGeolatvija(map) {

```

```
// atspējo ģeoportāla marķieru iznirstošos logus
map.EnableApplicationMarkerPopups(false);
}
mapAPI("geolatvija-map-modal", initGeolatvija);
</script>
```

## 4.3 Kartes notikumi

### 4.3.1 RegisterForEvent

Ļauj pierakstīties/sākt klausīties kādu no zemāk minētajiem notikuma veidiem.

Pirmajā parametrā jāpadod notikuma veids - "onMapClick", "onMarkerClick", "onApplicationMarkerClick" vai "onAddressClick".

Otrajā parametrā - funkcija, kas tiks izsaukta, iestājoties attiecīgajam notikumam. Atkarībā no notikuma veida funkcijai tiks padotas noteiktas vērtības, piemēram, "onMapClick" gadījumā koordinātas.

Katram notikuma veidam var pierakstīties tikai vienreiz.

Piemērā tiek attēlotas koordinātas vietai, kur lietotājs uz kartes ieklikšķina:

```
<script>
function initGeolatvija(map) {
    // pierēģistrējas "onMapClick"
    map.RegisterForEvent("onMapClick", function (longitude, latitude) {
        console.log({ longitude: longitude, latitude: latitude });
        alert("Koordināta: "+ longitude + " " + latitude);
    });
}
mapAPI("geolatvija-map-modal", initGeolatvija);
</script>
```



### 4.3.2 UnRegisterForEvent

Ļauj atreģistrēties / pārtraukt klausīties notikumu.

Parametrā jāpadod notikuma veids, kuru beigt klausīties.

Piemērā tiks attēlotas koordinātas tikai pirmajam klikšķim, jo pēc tā klausīšanās tiek pārtraukta:

```
<script>
function initGeolatvija(map) {
    map.RegisterForEvent("onMapClick", function (longitude, latitude) {
        console.log("onMapClick");
        console.log({ longitude: longitude, latitude: latitude });
        alert("Koordināta: "+ longitude + " " + latitude);
        console.log("UnRegisterForEvent for onMapClick");
        // beidz klausīties "onMapClick"
        map.UnRegisterForEvent("onMapClick");
    });
}
mapAPI("geolatvija-map-modal", initGeolatvija);
</script>
```

## 4.4. Pieejamie notikumu veidi

### 4.4.1 onMapClick

Ļauj iegūt LKS-92 jeb EPSG:3059 koordinātas vietai kartē, kur lietotājs ieklikšķina.

Funkcijai tiek atgrieztas divas vērtības - koordinātas garums un platums:

```
<script>
function initGeolatvija(map) {
    // pieregistrējas "onMapClick"
    // pēc klikšķa atgriež koordinātas
    map.RegisterForEvent("onMapClick", function (longitude, latitude) {
```

```
console.log({ longitude: longitude, latitude: latitude });
alert("Koordināta: "+ longitude + " " + latitude);
});
}
mapAPI("geolatvija-map-modal", initGeolatvija);
</script>
```

#### 4.4.2 onMarkerClick

Notikums, kas tiek izsaukts, kad lietotājs izvēlas API marķieri (klikšķina uz marķiera). Šis notikums apstrādā tikai ar API ("ShowMarkers") uz kartes izvietotos marķierus.

Notikumam iestājoties, funkcijai tiek padotas trīs vērtības:

- marķiera koordinātas ģeogrāfiskais garums;
- marķiera koordinātas ģeogrāfiskais platums;
- masīvs ar marķieriem (vienmēr saturēs tieši vienu marķieri), marķieris būs *JavaScript* objekts ar laukiem "COLOR" un "DESCRIPTION", kas tika norādīts marķierim "ShowMarkers" izsaukumā.

Piemērā pārlūka konsolē tiek ierakstītas funkcijas saņemtās vērtības, kad kartē tiek izvēlēts marķieris:

```
<script>
function initGeolatvija(map) {
  // izveido marķierus
  var markers = [
    { x: 506646, y: 312794, description: "Alberta iela 10, Rīga" },
    {
      x: 506686,
      y: 312764,
      description: '<div class="navlink"><a href="https://www.google.com/maps/search/?api=1&query="
target="_blank">How to get</a></div>',
      color: "rgba(170, 30, 200, 1.0)",
    },
  ],
```

```

];
// attēlo marķierus uz kartes
map.ShowMarkers(markers);
// iecentrē karti koordinātām, kur ir marķieri
map.ZoomToPoint(506646, 312794);
// klausās marķieru izvēli kartē
map.RegisterForEvent("onMarkerClick", function (x, y, markers) {
  console.log("onMarkerClick");
  console.log({ x, y, markers });
  //map.UnRegisterForEvent("onMarkerClick");
});
}
mapAPI("geolatvija-map-modal", initGeolatvija);
</script>

```

#### 4.4.3 onApplicationMarkerClick

Notikums, kas tiek izsaukts, kad lietotājs izvēlas ģeoportāla marķieri (klikšķina uz marķiera). Šis notikums apstrādā tikai marķierus, kas kartē izvietoti iegultās kartes izveidošanas laikā ģeoportālā.

Notikumam iestājoties, funkcijai tiek padotas trīs vērtības:

- marķiera koordinātas ģeogrāfiskais garums;
- marķiera koordinātas ģeogrāfiskais platums;
- masīvs ar marķieriem (vienmēr saturēs tieši vienu marķieri), marķieris būs *JavaScript* objekts ar laukiem "COLOR" (krāsa), "DESCRIPTION" (apraksts) un "UUID" (unikāls identifikators).

Piemērā pārlūka konsolē tiek ierakstītas funkcijas saņemtās vērtības, kad kartē tiek izvēlēts ģeoportāla marķieris:

```

<script>
function initGeolatvija(map) {
  // klausās ģeoportālā iegultās kartes veidošanas laikā izvietoto marķieru klikšķus

```

```
map.RegisterForEvent(
  "onApplicationMarkerClick",
  function (x, y, markers) {
    console.log("onApplicationMarkerClick");
    console.log({ x, y, markers });
    //map.UnRegisterForEvent("onApplicationMarkerClick");
  }
);
}
mapAPI("geolatvija-map-modal", initGeolatvija);
</script>
```

#### 4.4.4 onAddressClick

Ja kartē ir pievienots “Adrešu punktu” slānis (skat. “Izmantošanas priekšnoteikumi un ierobežojumi” sadaļu), tad šis notikums tiek izsaukts, kad lietotājs klikšķina uz adreses punkta.

Notikumam iestājoties, funkcijai tiek padotas trīs vērtības:

- klikšķa koordinātas ģeogrāfiskais garums;
- klikšķa koordinātas ģeogrāfiskais platums;
- masīvs ar visām adresēm, kas atrodas klikšķa vietā, adrese būs *JavaScript* objekts ar laukiem atbilstoši atvērto datu portālā publicētajiem Adrešu reģistra telpiskajiem datiem:  
<https://data.gov.lv/dati/lv/dataset/valsts-adresu-registra-informacijas-sistemas-atvertie-dati/resource/f539e8df-d4e4-4fc1-9f94-d25b662a4c38> , kur ģeometrija ir laukā “geometry” kā masīvs ar garumu un platumu un pilns adreses pieraksts laukā “std”.

Piemērā tiek attēlota pirmā no lietotāja izvēlētajām adresēm kartē:

```
<script>
function initGeolatvija(map) {
```

```
// klausās adrešu punktu klikšķus
map.RegisterForEvent("onAddressClick", function (x, y, addresses) {
  console.log("onAddressClick");
  console.log({ x, y, addresses });
  alert("Adrese: "+ addresses[0].std);
  //map.UnRegisterForEvent("onAddressClick");
});
}
mapAPI("geolativija-map-modal", initGeolativija);
</script>
```