

```
1 @page "/opladen-ev"
2 @attribute [StreamRendering]
3 @rendermode InteractiveServer
4 @inject IMeasurementRepository measurementRepository;
5
6 <PageTitle>Opaden EV</PageTitle>
7 <h1>Opaden EV (variant F)</h1>
8 <p>
9     Hier vindt u gedetailleerde informatie over de energieverbruikspatronen en
10    de kosten van het opladen van elektrische voertuigen.
11    De gegevens worden in real-time bijgewerkt, zodat u altijd over de meest
12    actuele informatie beschikt.
13    Daarnaast kunt u de tabel bekijken die inzicht geeft in laadsessies over
14    verschillende tijdsperioden.
15    Dit stelt u in staat om trends te herkennen en uw energieverbruik
16    efficiënter te beheren.
17    <br /><br />
18    <button class="btn" style="text-decoration: underline; text-decoration-
19    color: blue;" data-bs-toggle="modal" data-bs-target="#AlgoritmeUitleg">Meer
20    informatie over het algoritme</button>
21 </p>
22 <div class="modal fade" id="AlgoritmeUitleg" tabindex="-1" aria-
23 labelledby="exampleModalLabel" aria-hidden="true">
24     <div class="modal-dialog">
25         <div class="modal-content">
26             <div class="modal-header">
27                 <h1 class="modal-title fs-5" id="exampleModalLabel">Algoritme
28                 informatie</h1>
29                 <button type="button" class="btn-close" data-bs-dismiss="modal" aria-
30                 label="Close"></button>
31             </div>
32             <div class="modal-body">
33                 <P>Dit algoritme detecteert laadsessies op basis van
34                 energieverbruikdata. Het werkt als volgt:</P>
35
36                 <ul>
37                     <li>Data Inlezen: Meetgegevens worden opgehaald met een bepaald
38                     aggregatie-interval.</li><br />
39                     <li>Laadsessie Start: Een laadsessie start wanneer het verbruik een
40                     ingestelde drempelwaarde overschrijdt.</li><br />
41                     <li>Verbruik en Kosten: Tijdens de sessie wordt het totale verbruik
42                     in kWh en de kosten berekend. Kosten kunnen dynamisch of vast zijn.</li><br />
43                     <li>Laadsessie Stop: De sessie eindigt wanneer het verbruik onder
44                     de drempel daalt en wordt alleen opgeslagen als de duur langer is dan 5
45                     minuten.</li><br />
46                 </ul>
47                 <p>Het algoritme slaat alleen laadsessies op die voldoen aan de
48                 minimumduur en rondt de resultaten af voor opslag.</p>
49             </div>
50             <div class="modal-footer">
51                 <button type="button" class="btn btn-secondary" data-bs-
52                 dismiss="modal">Close</button>
```

```
36     </div>
37   </div>
38 </div>
39 </div>
40 <hr>
41 <div class="main-content" style="width: 100%;">
42   <div class="laadsessies" style="margin-top: 0.5rem; width: 100%;">
43     <div class="function-row-container" style="display: flex; align-items:
44 center; text-align: center; justify-content: space-between; gap: 6rem;">
45       <div class="input-row-container" style="display: flex; height: 4rem;
46 gap: 1rem;">
47         <div class="form-floating mb-3" style="width: 11rem; height: 4rem;">
48           <input type="number" min="3500" style="height: 4rem;" class="form-
49 control" id="floatingInputValue" placeholder="3500" aria-label="3500" aria-
50 describedby="button-addon2" @bind="detectieVermogenW">
51           <label for="floatingInputValue">Detectiepunt (Watt)</label>
52         </div>
53         <div class="form-floating mb-3" style="width: 11rem; height: 4rem;">
54           <input type="number" min="1" max="30" style="height: 4rem;"
55 class="form-control" id="floatingInputValue" placeholder="3500" aria-
56 label="3500" aria-describedby="button-addon2" @bind="numberOfDays">
57           <label for="floatingInputValue">Aantal dagen</label>
58         </div>
59         <div class="form-floating mb-3" style="width: 11rem; height: 4rem;">
60           <input type="number" min="0" style="height: 4rem;" class="form-
61 control" id="floatingInputValue" placeholder="5" aria-label="5" aria-
62 describedby="button-addon2" @bind="minimumduur">
63           <label for="floatingInputValue">minimumduur (minuten)</label>
64         </div>
65         <div class="form-floating mb-3" style="width: 11rem; height: 4rem;">
66           <input type="number" min="0" style="height: 4rem;" class="form-
67 control" id="floatingInputValue" placeholder="0,25" aria-label="0,25" aria-
68 describedby="button-addon2" @bind="prijsCatagorieVast">
69           <label for="floatingInputValue">Prijs per kWh (Euro)</label>
70         </div>
71       </div>
72       <div class="button-row-container" style="display: flex; height: 3rem;
73 gap: 1rem;" >
74         <div class="btn-group" role="group" aria-label="Basic radio toggle
75 button group">
76           <input type="radio" class="btn-check" name="btnradio"
77 id="btnradio2" autocomplete="off" checked />
78           <label class="btn btn-outline-primary d-flex justify-content-center
79 align-items-center" for="btnradio2" @onclick="() => prijsCategorieDynamisch =
80 false" checked>Vast</label>
81           <input type="radio" class="btn-check" name="btnradio"
82 id="btnradio1" autocomplete="off" />
83           <label class="btn btn-outline-primary d-flex justify-content-center
84 align-items-center" for="btnradio1" @onclick="() => prijsCategorieDynamisch =
85 true">Dynamisch</label>
86         </div>
87         <button class="btn btn-primary" type="submit"
88 @onclick="LoadDataAsync">Refresh</button>
89       </div>
90     </div>
91   </div>
92 </div>
```

```

72     @if (currentPowerData ≠ null)
73     {
74         @if(Laadsessies.Count = 0 || Laadsessies = null)
75         {
76             <p style="margin-top: 2rem;">Helaas geen laadsessies beschikbaar.
Probeer de pagina opnieuw te laden!</p>
77         }
78         else
79         {<div style="margin-top: 1rem;">
80             <table class="table">
81                 <thead>
82                     <tr>
83                         <th>Start</th>
84                         <th>Stop</th>
85                         <th>Duration</th>
86                         <th>Used kWh</th>
87                         <th>Price</th>
88                     </tr>
89                 </thead>
90                 @foreach (Laadsessie sessie in Laadsessies)
91                 {
92                     <tbody>
93                         <tr>
94                             <td>@sessie.Start</td>
95                             <td>@sessie.Stop</td>
96                             <td>@sessie.Duur</td>
97                             <td>@sessie.TotaalKWh.ToString("F2") kWh</td>
98                             <td>@sessie.Prijs.ToString("C")</td>
99                         </tr>
100                     </tbody>
101                 }
102             </table>
103         </div>
104     }
105 }
106 }
107 else
108 {
109     <div style="display: flex; flex-direction: column; justify-content:
center; align-items: center; text-align: center; width: 100%;">
110         <p style="margin-top: 1rem;">Calculating</p>
111         <div class="spinner-border text-dark" role="status">
112             <span class="visually-hidden">Data wordt opgehaald ... </span>
113         </div>
114     </div>
115 }
116 </div>
117 </div>
118
119
120 @code {
121     // Variable for database connection
122     double detectieVermogenW = 3500D;
123     int meterId = 10019163;
124     int numberOfDays = 30;

```

```
125     string aggregationWindow = "1m";
126     private List<Measurement>? currentPowerData;
127
128     // Variable for LaadsessieDetectie
129     int minimumduur = 5;
130     double prijsCatagorieVast = 0.25D;
131     bool prijsCategorieDynamisch = false;
132     private List<Laadsessie> Laadsessies = new List<Laadsessie>();
133
134     protected override async Task OnInitializedAsync()
135     {
136         await LoadDataAsync();
137     }
138
139     private async Task LoadDataAsync()
140     {
141         Laadsessies = new List<Laadsessie>();
142         bool laadsessieActief = default;
143         double totaalkWhLaadsessie = default;
144         double prijsLaadsessie = default;
145         TimeSpan duurLaadsessie = default;
146         DateTime startLaadsessie = default;
147         DateTime stopLaadsessie = default;
148         currentPowerData = await this.measurementRepository.GetPowerUsed(meterId,
149             numberOfDays, aggregationWindow);
150
151         // Algoritme laadsessie detectie
152         foreach(var measurement in currentPowerData) {
153             if(measurement.Value.Value > detectieVermogenW && laadsessieActief ==
154                 false)
155             {
156                 startLaadsessie = measurement.Timestamp;
157                 laadsessieActief = true;
158                 totaalkWhLaadsessie += measurement.Value.Value * (1D / 60D) / 1000D;
159                 prijsLaadsessie += measurement.EnergyPrice.Value *
160                     (measurement.Value.Value * (1D / 60D) / 1000D);
161             }
162             else if(measurement.Value.Value > detectieVermogenW && laadsessieActief
163                 == true)
164             {
165                 totaalkWhLaadsessie += measurement.Value.Value * (1D / 60D) / 1000D;
166                 prijsLaadsessie += measurement.EnergyPrice.Value *
167                     (measurement.Value.Value * (1D / 60D) / 1000D);
168             }
169             else if(measurement.Value < detectieVermogenW && laadsessieActief ==
170                 true)
171             {
172                 stopLaadsessie = measurement.Timestamp;
173                 duurLaadsessie = stopLaadsessie.Subtract(startLaadsessie);
174
175                 // Calculate prijs met vaste prijs
176                 if(prijsCategorieDynamisch == false)
177                 {
```

```
174     prijsLaadsessie = totaalkWhLaadsessie * prijsCatagorieVast;
175 }
176
177     prijsLaadsessie = Math.Round(prijsLaadsessie, 2);
178     totaalkWhLaadsessie = Math.Round(totaalkWhLaadsessie, 2);
179
180     // Check minimum duration
181     if(duurLaadsessie.TotalMinutes ≥ minimumduur)
182     {
183         var singleLaadsessie = new Laadsessie(startLaadsessie,
184 stopLaadsessie, duurLaadsessie, totaalkWhLaadsessie, prijsLaadsessie);
185         Laadsessies.Add(singleLaadsessie);
186     }
187
188     laadsessieActief = default;
189     totaalkWhLaadsessie = default;
190     prijsLaadsessie = default;
191     duurLaadsessie = default;
192     startLaadsessie = default;
193     stopLaadsessie = default;
194 }
195
196     else {
197         continue;
198     }
199 }
200     await InvokeAsync(StateHasChanged);
201 }
```