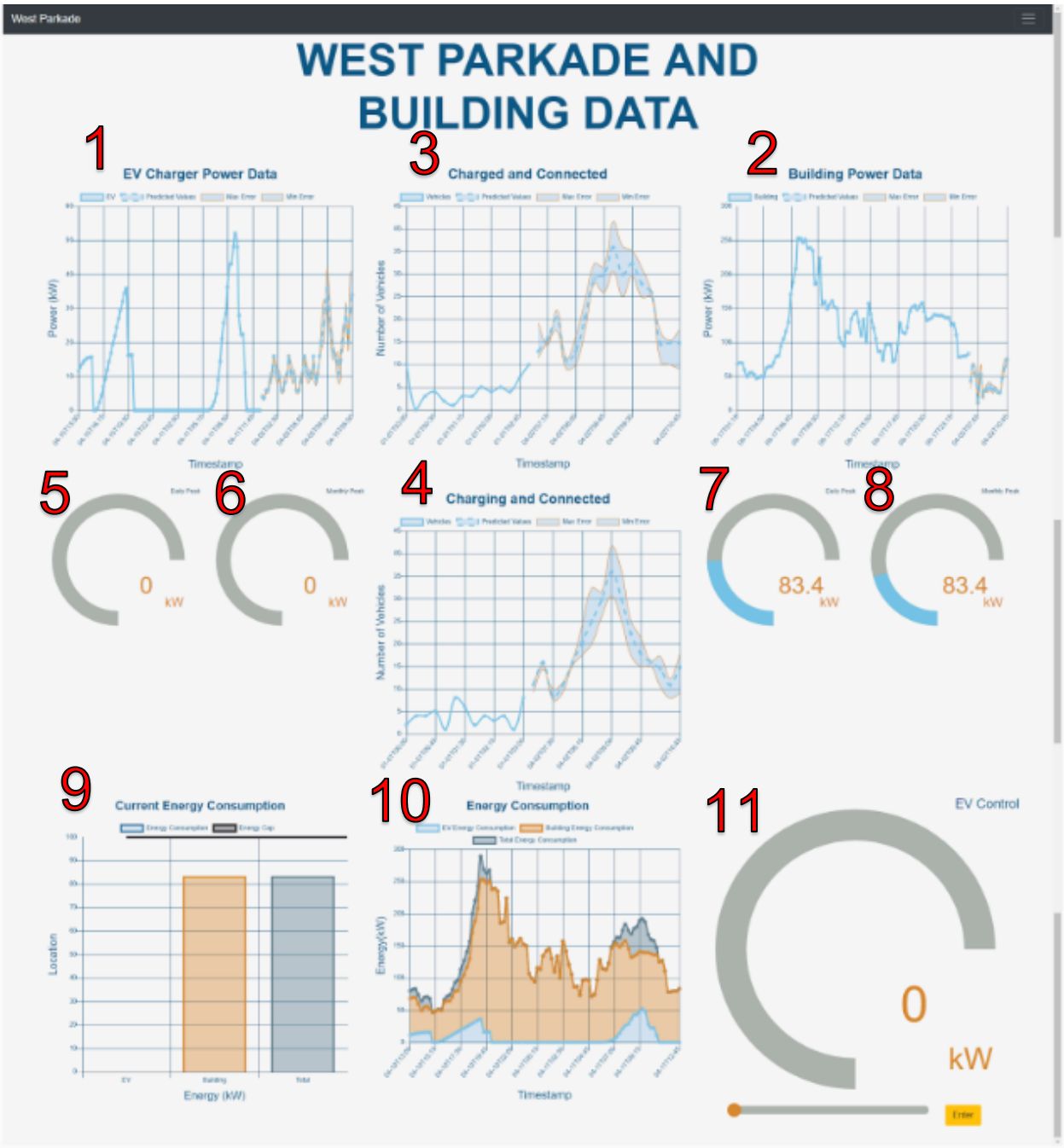
West Parkade
WEST PARKADE AND BUILDING DATA
1
EV Charger Power Data
Power (kW)
Timestamp
3
Charged and Connected
Number of Vehicles
Timestamp
2
Building Power Data
Power (kW)
Timestamp
5
0
kW
6
0
kW
4
Charging and Connected
Number of Vehicles
Timestamp
7
83.4
kW
8
83.4
kW
9
Current Energy Consumption
Location
Energy (kW)
10
Energy Consumption
Energy(kW)
Timestamp
11
EV Control
0
kW
Enter

1. Realtime EV data
    - To modify where to GET the real time/historic data, in the HTML template {% block jquery %}, change the **var evpoint = {new url}**
    - To modify where to GET the predicted data, in the HTML template {% block jquery %}, change the **var evpredictedpoint = {new url}**
    - To edit the charts labels, in the HTML template {% block jquery %}, edit the last four variables in **var evChart = createChart(*args, label title, x-axis label, y-axis label, label for the main data series)**

```
/**
 * Creates a chart for either building or ev along with predicted
values.
 *
 * @param {Array} dates Array of dates for our chart.
 * @param {Array} pwr_vals Array of power for our chart.
 * @param {Array} future_pwr Array of future values for our chart.
 * @param {Array} maxerr_pwr Array of max error for future values.
 * @param {Array} minerr_pwr Array of min error for future values.
 * @param {string} id HTML element id.
 * @param {string} lbl_title Title of the chart.
 * @param {string} x_axis Title for x-axis of the chart.
 * @param {string} y_axis Title for y-axis of the chart.
 * @param {string} _lbl Label for our main data series.
 *
 * @returns {Chart} Chart object.
 */
function createChart(dates, pwr_vals, future_pwr, maxerr_pwr,
minerr_pwr, id, _title, x_axis, y_axis, _lbl)
```

2. Realtime Building data
    - Similar to 1, except modify the following variables:
    - **var buildingpoint**
    - **var bdpredictedpoint**
    - **var bdChart = createChart()**
3. Realtime Charged (finished) and Connected Vehicle data
    - Similar to 1, except modify the following variables:
    - **var chargedcarpoint**
    - **var chargedpredictedpoint**
    - **var chargedcarChart = createChart()**
4. Realtime Charging and Connected Vehicle data
    - Similar to 1, except modify the following variables:
    - **var chargingcarpoint**
    - **var chargingpredictedpoint**
    - **var chargingcarChart = createChart()**
5. Realtime Daily EV gauge
    - To modify our gauge, in the HTML template {% block content %} there is a element with **<div id="evdailygauge" …. > …. **

- To modify our labels
    - To change the units (ie. kW to kWh) modify the **units here**
    - To change the label modify the **label here**
- To change the current value, modify the **<input type="hidden" id="evdailyval" value={{curr_ev}}>**, note that the **{{curr_ev}}** is what's being sent from *views.graph.getgraph()* by that name.
- To change the maximum value, modify the **<input type="hidden" id="evdailymax" value={{max_evdaily}}>**, similarly note the **{{max_evdaily}}** is the name sent from *views.graph.getgraph()*.
- Note: If you change the **id** of any of the elements, you will need to find that element id in the rest of the template and change it accordingly as well, particularly in the **createGauge()** in the {% block jquery %}

```
/**
 *
 * @param {double} val_id HTML id where our value gauge is set to.
 * @param {double} maxval_id HTML id for our Maximum value of gauge.
 * @param {string} id HTML ID for our gauge element.
 *
 * @returns {Gauge} returns a Gauge object.
 */
function createGauge(val_id, maxval_id, id)
```

6. Realtime Monthly EV gauge
    - Similar to 5, but with the following variables:
    - ** **
    - **<input type="hidden" id="evmonthlyval" value={{curr_ev}}>**
    - **<input type="hidden" id="evmonthlymax" value={{max_evmonthly}}>**
    - Labels are the same just in the corresponding element
7. Realtime Daily Building gauge
    - Similar to 5, but with the following variables:
    - ** **
    - **<input type="hidden" id="bddailyval" value={{curr_bd}}>**
    - **<input type="hidden" id="bddailymax" value={{max_bddaily}}>**
    - Labels are the same just in the corresponding element
8. Realtime Monthly Building gauge
    - Similar to 5, but with the following variables:
    - ** **
    - **<input type="hidden" id="bdmonthlyval" value={{curr_bd}}>**
    - **<input type="hidden" id="bdmonthlymax" value={{max_bdmonthly}}>**
    - Labels are the same just in the corresponding element
9. Realtime Bar chart
    - To modify the bar chart, in the HTML template {% block jquery %} the **var barChart = createBarChart()**

- Most importantly is the parameter *powercap*, which is what the line will be set to

```
/**
 * Creates a bar chart for the power consumption of EV, Building and
total.
 *
 * @param {Array} locations Array of locations for our chart.
 * @param {Array} values Values corresponding to power at each
location.
 * @param {double} powercap A cap value that you wish to avoid
hitting.
 * @param {string} id Element ID in HTML.
 * @param {string} title Title of our chart.
 * @param {string} x_axis Label for our x_axis.
 * @param {string} y_axis Label for our y_axis.
 *
 * @returns {Chart} Returns a chart object.
 */
function createBarChart(locations, values, powercap, id, title,
x_axis, y_axis)
```

10. Realtime Combination chart
    - To modify the combination chart, in the HTML template {% block jquery %} the **var testChart = createComboChart()**

```
/**
 * Creates a chart with EV power, Building Power, and Total Power.
 * Assumes that the datetime for EV and Building power are the same.
 *
 * @param {Array} dates array of datetime.
 * @param {Array} ev_pwr array of ev power values.
 * @param {Array} bd_pwr array of building power values.
 * @param {string} id Element id in HTml.
 * @param {string} title title of the chart.
 * @param {string} x_axis label of x-axis.
 * @param {string} y_axis label of y-axis.
 *
 * @returns {Chart} returns a chart object.
 */
function createComboChart(dates, ev_pwr, bd_pwr, id, title, x_axis,
y_axis)
```

11. Control gauge
    - Similar to 5, but with the following variables for the gauge
    - ** **
    - Labels:

    - **units here**
    - **label here**
- **<input type="hidden" id="evcurrval" value={{curr_ev}}>**
- **<input type="hidden" id="evmaxval" value="200">**, change this according to however you want to set the max value into
- The slider variables are found just below the control gauge element in HTML
- **<input type="range" min="0" max="200" step="0.001" value={{curr_ev}} class="slider" id="TestEV">** is the slider
- **<button type="button" class="btn btn-warning">Enter</button>** corresponds to the button
- In the {% block jquery %} near the end, there is a function that controls the gauge and slider interaction, **evslider.oninput = function () { evctrlgauge.setValue(this.value) }**