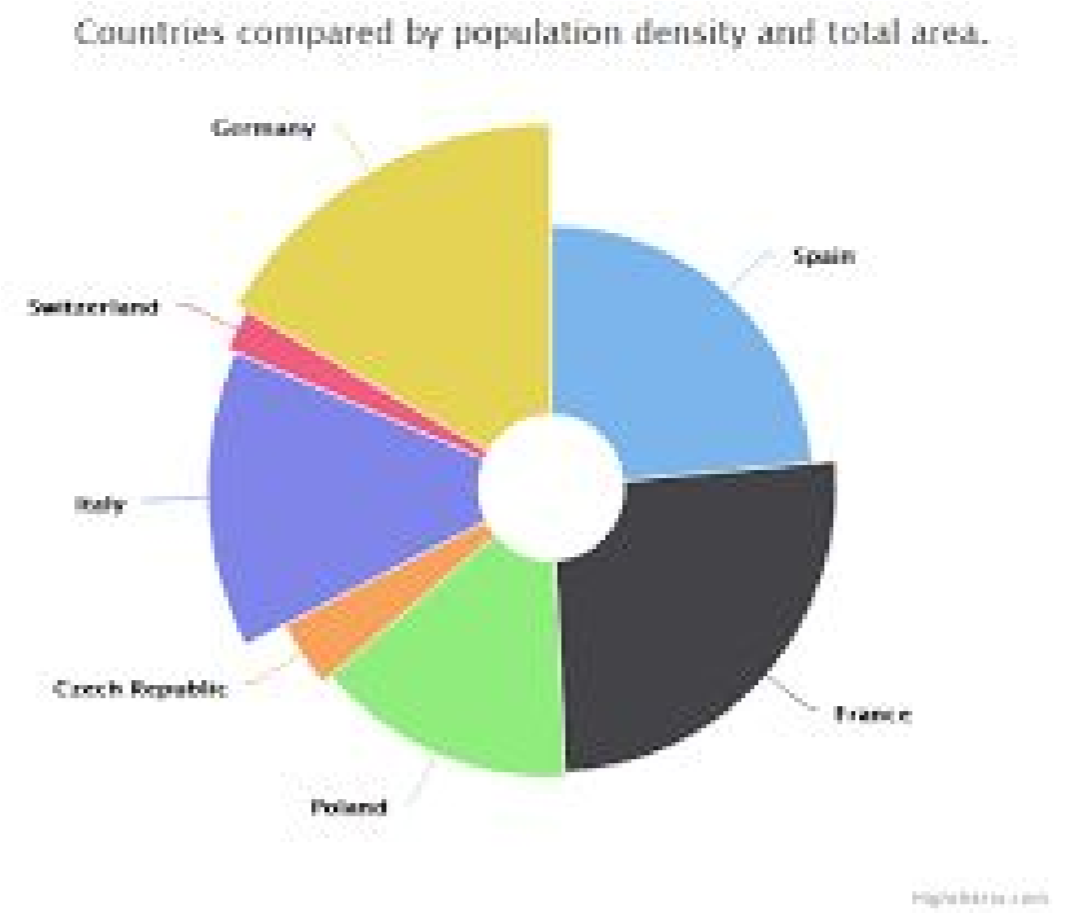
**WeGot Data Visuals**

**With the Request and Response JSONs**

# Pie Chart:



**Fetch the quantity of water from different Sources in the site for the given date range.**

Request:

{

“fromDate”:”10-01-2018”,

“toDate”:”14-02-2018”,

“uid”:”userID”,

“chartType”:”pie”

}

Response:

[

{name: 'Source1', value: 61.41 },

{name: 'Source2', value: 11.84 },

{name: 'Source3', value: 10.85 },

{name: 'Source4', value: 12.85 }

]

## Queries:

1. What is uid represents? Is Any formats exists?

1. how to find site details like
   1. Apartment Name -
   2. Block Name -
   3. Flat/Door No -
2. w2\_water\_type vs w2\_water\_source\_type tables in DB??

Source tyes

1. Pie Chart for a specific flat or apartment?

aartment

1. Cannot found table to fetch volume of water used from each water source type.

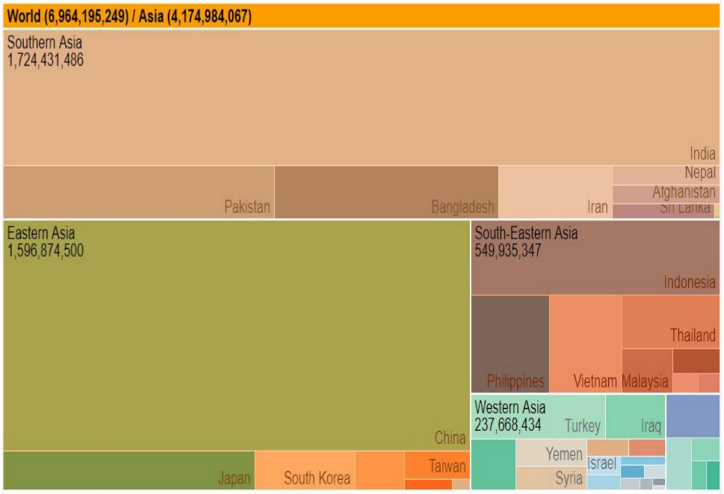
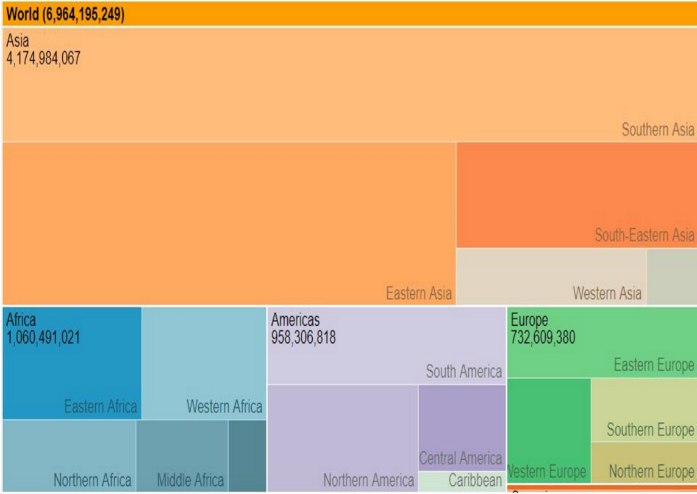
## Report app to DB calls:

1. To fetch all available water sources and their id’s.

[**SELECT**](http://54.229.208.9/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/select.html)**\* FROM w2\_water\_source\_type**

1. Fetch volume of water used from each water source type for the given date range.
2. Prepare JSON to send based result set from above 2 calls.

# Tree Map:



**Fetch the individual residence consumption in their respective blocks for the given date range.**

Request:

{

“fromDate”:”10-01-2018”,

“toDate”:”14-02-2018”,

“uid”:”userID”,

“chartType”:”TreeMap”

}

Response:

[

{ "key": "BlockA", "region": "House1", "value": 25500100 },

{ "key": "BlockA", "region": "House2", "value": 1231231 },

{ "key": "BlockB", "region": "House1", "value": 1231231 }

]

## Queries:

How to identify block and apartment id from UID? If not available in UID then how to get from DB?

Is Tree chart represents all the blocks in an apartment?

All blocks

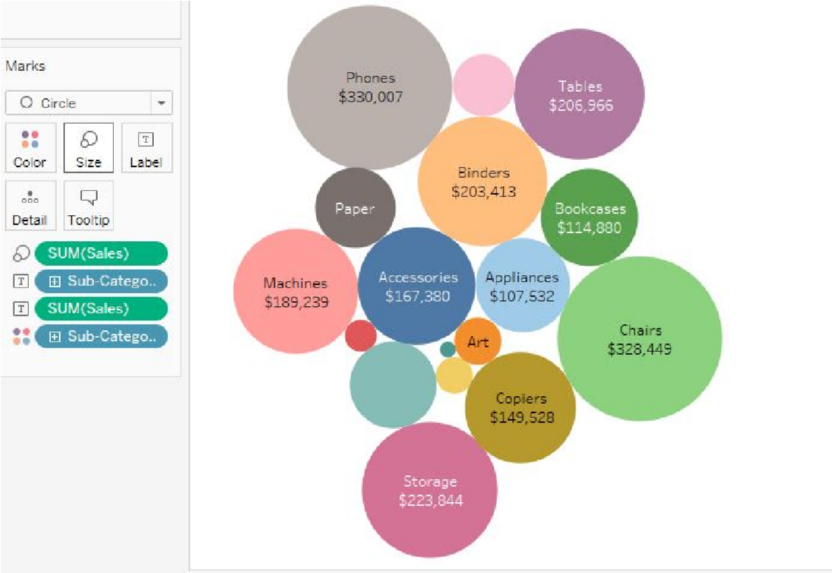
## Report app to DB calls:

Get Block Id and Apartment Id

Get No of House for a block

Get individual house consumption

# Bubble Chart:



**Fetch the house numbers with their consumption values to highlight the high consumers for a given date range.**

Request:

{

“fromDate”:”10-01-2018”,

“toDate”:”14-02-2018”,

“uid”:”userID”,

“chartType”:”BubbleChart”

}

Response:

[

{"id":"BlockA-House1","value":"200"},

{"id":"BlockA-House2","value":"300"},

{"id":"BlockB-House3","value":"400"},

{"id":"BlockC-House4","value":500}

]

## Queries:

## Report app to DB calls:

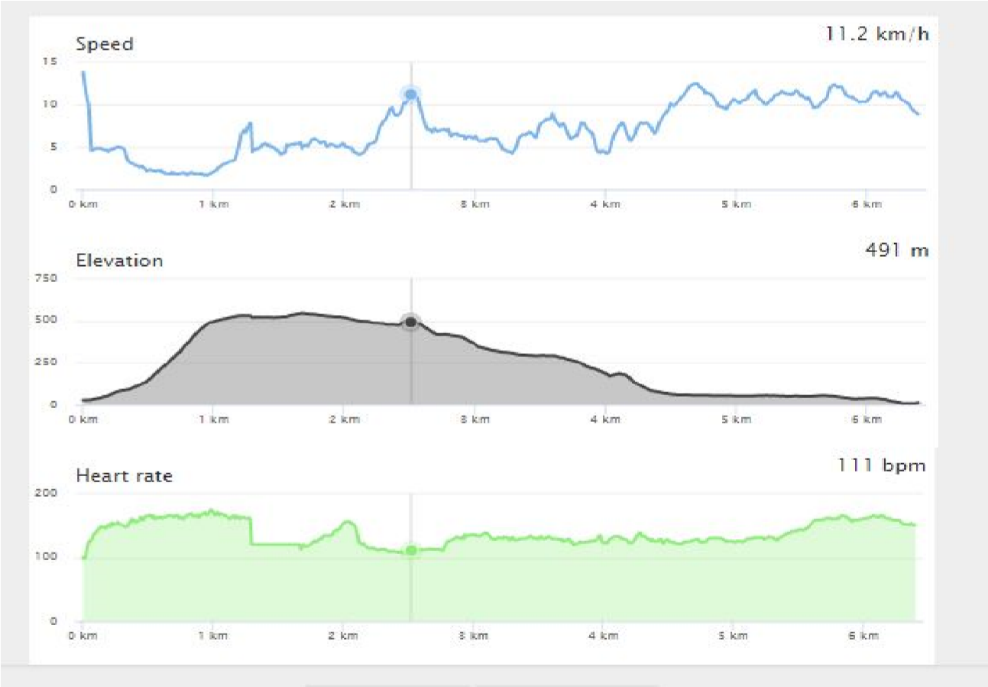
Get Apartment Id

Get Available blocks

Get No of House for a block

Get individual house consumption

# Raw/Treated/STP Comparison:



**Fetch the three different varieties of water with their demand on an hourly basis for a given date range.**

Request:

{

“fromDate”:”10-01-2018”,

“toDate”:”14-02-2018”,

“fromTime”:”9AM”,

“toTime”:”12PM”

“chartType”:”RTS”

}

Response:

Below is the example:

xData – It is the log timestamp and data – Values

{

"xData": [09:00,09:10,09:20,09:40,10:15,10:45,11:00,11:45,11:50,12:00],

"datasets": [{

"name": "Raw",

"data": [833,524,441,651,961,566,617,728,823,844],

"unit": "kl",

"type": "area",

"valueDecimals": 1

}, {

"name": "Treated",

"data": [126,217,217,27,127,130,32,133,35,136],

"unit": "kl",

"type": "area",

"valueDecimals": 0

}, {

"name": "STP",

"data": [101,98,103,115,124,128,133,138,138,141],

"unit": "kl",

"type": "area",

"valueDecimals": 0

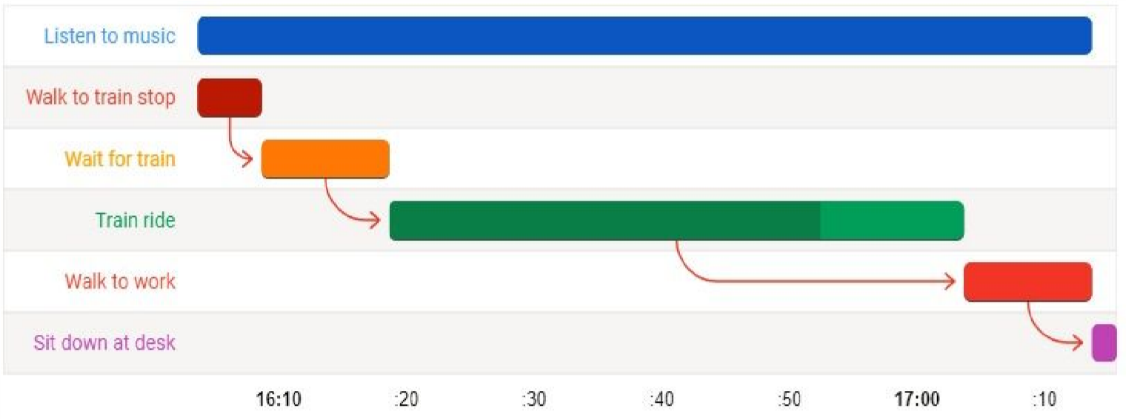
}]

}

Sample taken from:

<https://cdn.rawgit.com/highcharts/highcharts/057b672172ccc6c08fe7dbb27fc17ebca3f5b770/samples/data/activity.json>

# Gantt chart:



**Fetch the ON/OFF states of each pump for a given date and their yield during the ON period.**

Request:

{

“date”:”10-01-2018”,

“uid”:”userID”,

“chartType”:”ganttchart”

}

Response:

[

{”source”:”pump1”, start: 123123, end: 12313123123,”yield”:”2”},

{”source”:”pump2”, start: 86400000, end: 123400000,”yield”:”4”}

{”source”:”pump3”, start: 86445560, end: 12434000,”yield”:”5”}

]

Note: Start / End – Timestamp

## Queries:

Chart shows pumps data for user’s corresponding block (pumps installed in that block) or the entire site (all the pumps in the site)?

**Entire ste**

W2\_pump\_status\_log vs w2\_pump\_state

## Report app to DB calls:

Get apartment Id

Get pumps available in the apartment and their Id

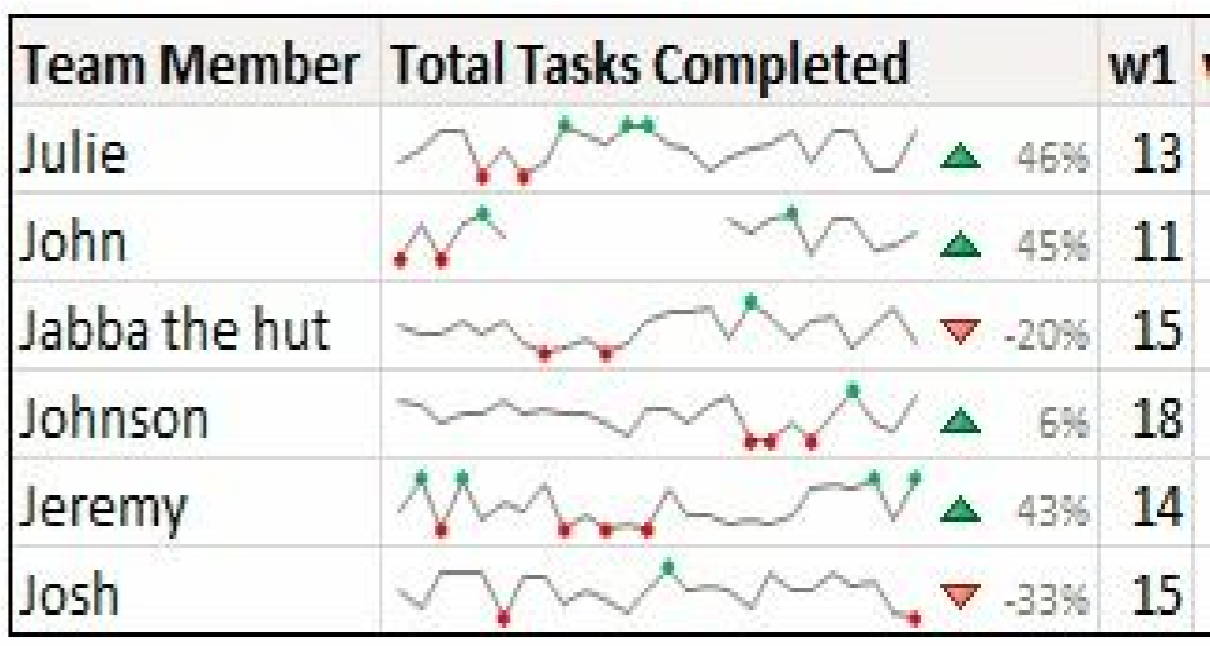
[**SELECT**](http://54.229.208.9/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/select.html)**\* FROM w2\_pumps**

Get each pump’s on/off timestamp for a given day

[**SELECT**](http://54.229.208.9/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/select.html)**\* FROM w2\_pump\_state or** [**SELECT**](http://54.229.208.9/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/select.html)**\* FROM W2\_pump\_status\_log**

Get yield of a pump for the above timestamps

# Sparkline Chart:



**Compares all the sources and their performances for the given week in terms of their yield**

Request:

{

“fromdate”:”10-01-2018”,

“todate”:”20-01-2018”,

“uid”:”userID”,

“chartType”:”sparkline”

}

Response:

{

"data": [{

"name": "Source1",

"SLData": [12,32,45,0,23],

"bar": [32,0,23,53,45,77,0],

"yeildValue": 232,

"yeildPercentage": 88

},

{

"name": "Source2",

"SLData": [43,37,99,55,0],

"bar": [32,0,23,53,45,77,0],

"yeildValue": 232,

"yeildPercentage": 88

}]

}

# Heat Map:

****

**Fetches the consumption of water for the given date as a heat map.**

Request:

{

“fromdate”:”10-01-2018”,

“todate”:”20-05-2018”,

“uid”:”userID”,

“chartType”:”heatmap”

}

Response:

{

"data": [{

"xValue": "JAN 2017",

"yValue": "01",

"Data": 12312312

},

{

"xValue": "JAN 2017",

"yValue": "02",

"Data": 43234

},

{

"xValue": "JAN 2017",

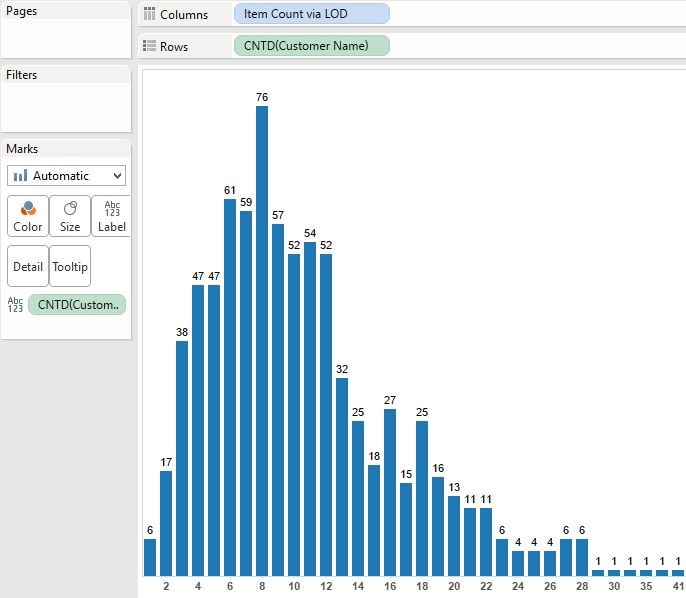
"yValue": "03",

"Data": 4356354

}]

}

# Follow UP Histogram:



**Fetches the consumption for the selected date in the heat map.**

Request:

{

“date”:”10-01-2018”,

“uid”:”userID”,

“chartType”:”heatmap”

}

Response:

[{

"name": "2018-05-10",

"data": [{

"bin": 9:00,

"count": 30000

},

{

"bin": 10:00,

"count": 80000

},

{

"bin": 11:00,

"count": 180000

},

{

"bin": 12:00,

"count": 40000

},

{

"bin": 13:00,

"count": 40000

},

{

"bin": 14:00,

"count": 40000

}]

}]