



SMART PLUG

Productiedossier

Analyse	2
1. Algemene doelstelling	2
2. Technische scope	2
3. Functionele scope	4
• Dashboard	4
• Stats	4
• Actions	4
• User Panel	4
Wireflow	5
Code snippets	6
Bijlage: timesheet	7

Analyse

1. Algemene doelstelling

Het doel van deze app was om mensen meer bewust te maken over hun eigen energiegebruik. Als het bv. niet nodig is om onnodige energie te verbruiken kan de gebruiker het apparaat dat verantwoordelijk is voor het gebruik via de app regelen. Het resultaat is daarom om zoveel mogelijk onnodige energie te vermijden.

2. Technische scope

De responsive mobile web app is ontwikkeld met behulp van de volgende programma's:

- Adobe Creative Cloud (PS, AI,...)
- Sequel pro
- Tower 2
- Postman
- MAMP PRO
- Firefox Developer Edition / Chrome
- Visual Studio Code

Voor de serverside is gebruik gemaakt van de volgende technologieën:

- Laravel / PHP
- MySQL Database
- PHP
- Firebase Realtime database
- Python
- Raspberry PI
- sensor 18B20

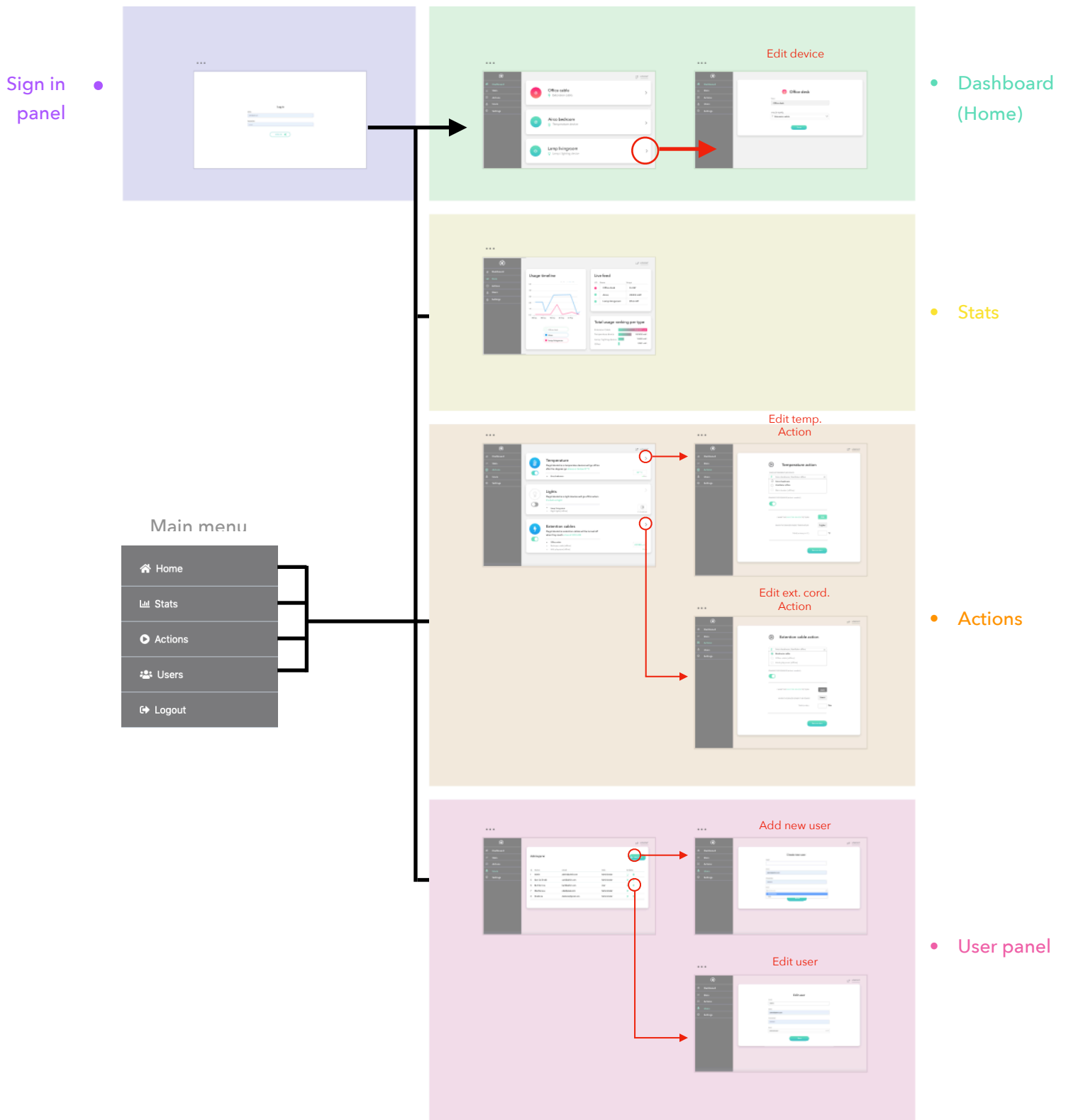
Ten slotte is voor de client side gebruik gemaakt van de volgende technologieën:

- Vue.js
- Node
- RESTful API
- Tailwind.js
- HTML5 / Javascript

3. Functionele scope

- **Dashboard**
 - Lijst met alle verbonden apparaten
 - Status kan manueel worden aangepast (aan/uit)
 - Device kan worden gepersonaliseerd
name & type device (temperature, lamps, extension cord, of other device)
 - Na het personaliseren wordt er een icoon weergegeven dat het type apparaat aanduidt en/of een aangepaste naam
- **Stats**
 - De gebruiker kan twee grafieken zien :
 - een real time teller van het verbruik oer apparatuur
 - Het gebruik van elk apparaat weergegeven op een interactieve kaart
- **Actions**
 - momenteel zijn er 3 "actions" beschikbaar:
 - voor de apparaten met het type 'temperatuur'
 - voor de apparaten met het type 'lights' bv. Een lamp
 - voor de apparaten met het type 'extension cable'
 - Het is de bedoeling dat er voor elk apparaat waarden kunnen worden ingesteld waardoor er een trigger wordt afgevuurd zodat het apparaat uit of aan kan worden gezet (bv. Als de temperatuur boven de 30°C is worden een triggers gestuurd naar de geselecteerde apparaten waardoor die "aan" gezet worden
- **User Panel**
 - De user panel is een ophijsting van alle users en hun rol men kan:
 - Een gebruiker aanmaken
 - Een gebruiker bewerken
 - Een gebruiker verwijderen
 - Ook is het mogelijk om aan elke gebruiker een rol toe te wijzen:
 - Een user (beperkte machtiging)
 - De administrator (onbeperkte bemachtiging)

Wireflow



Code snippets

```
freshUsage() {
  Outlet.getToken().then(tokenRes => {
    Outlet.outletGetList(tokenRes.data.result.token).then(resDevices => {
      let arrDevices = resDevices.data.result.deviceList;

      for (let i = 0; i < this.outlets.length; i++) {
        for (let o = 0; o < arrDevices.length; o++) {
          if (arrDevices[o].deviceId == this.outlets[i].device_id) {
            Outlet.outletUsage(
              tokenRes.data.result.token,
              arrDevices[o].deviceId
            ).then(resUsage => {
              let toParse = JSON.parse(resUsage.data.result.responseData);
              this.outlets[i].emeter.power_mw =
                toParse.emeter.get_realtime.power_mw;
            });
          }
        }
      }
    });
  });
},

usageTable() {
  Outlet.getToken().then(tokenRes => {
    Outlet.outletGetList(tokenRes.data.result.token).then(resDevices => {
      let arrDevices = resDevices.data.result.deviceList;

      axios
        .request({
          baseURL: "http://localhost:8000/api/v1/devices",
          method: "get"
        })
        .then(res => {
          if (res.data.length) {
            for (let i = 0; i < arrDevices.length; i++) {
              let outletObj = {
                device_id: arrDevices[i].deviceId,
                name: arrDevices[i].alias,
                status: 0,
                emeter: {}
              };

              for (let o = 0; o < Object.keys(res.data).length; o++) {
                if (arrDevices[i].deviceId == res.data[o].device_id) {
                  outletObj.name = res.data[o].alias;
                }
              }
            }
          }
        })
      }
    });
  });
},
```

```
async getToken() {
  let authEmail = process.env.TP_LINK_EMAIL;
  let authPass = process.env.TP_LINK_PASS;

  return axios.request({
    baseURL: 'https://wap.tplinkcloud.com',
    method: 'post',
    data: {
      "method": "login",
      "params": {
        "appType": "Kasa_Android",
        "cloudUserName": authEmail,
        "cloudPassword": authPass,
        "terminalUUID": "2adb9a64-3371-47e8-b1aa-cb78abb0d301"
      }
    }
  });
}

async outletGetList(tokenRes) {
  return axios.request({
    baseURL: 'https://eu-wap.tplinkcloud.com/?token=' + tokenRes,
    method: 'post',
    data: {
      "method": "getDeviceList"
    }
  });
}

async outletInfo(tokenRes, deviceId) {
  return axios.request({
    baseURL: 'https://eu-wap.tplinkcloud.com/?token=' + tokenRes,
    method: 'post',
    data: {
      "method": "passthrough",
      "params": {
        "deviceId": deviceId,
        "requestData": "{\\\"system\\\":{\\\"get_sysinfo\\\":null\\\"}}",
      }
    }
  });
}
```

```
{
  path: '/admin/users/edit/:id',
  component: editUser,
  name: 'admin.users.edit',
  meta: {
    authUser: false,
  }
},
{
  path: '/admin/stats',
  component: stats,
  name: 'stats',
  meta: {
    authUser: false,
  }
},
{
  path: '/actions',
  component: actions,
  name: 'actions',
  meta: {
    authUser: true,
  }
}

port default new VueRouter({
  mode: 'history',
  routes
})
```

```
var let todayDate = new Date();
re let thisMonth = todayDate.getMonth() + 1;
); let todayDate: Date;
let todayDate = new Date();
let thisMonth = todayDate.getMonth() + 1;
let thisYear = todayDate.getFullYear();

Outlet.getToken().then(tokenRes => {
  Outlet.outletGetList(tokenRes.data.result.token)
    .then(resDevices => {
      let arrDevices = resDevices.data.result.deviceList;

      for (let u = 0; u < arrDevices.length; u++) {
        Outlet.outletDaystat(
          tokenRes.data.result.token,
          arrDevices[u].deviceId,
          thisMonth,
          thisYear
        ).then(resUsage => {
          let toParse = JSON.parse(resUsage.data.result.responseData);
          let dayList = toParse.emeter.get_daystat.day_list;
          let xMonthArray = [];

          for (let i = 1; i <= getDaysInMonth(thisMonth, thisYear); i++) {
            if (dayList[i]) {
              if (
                dayList[i].month == thisMonth &&
                dayList[i].year == thisYear
              ) {
                if (u == 0) {
                  let dayString = String(
                    dayList[i].year +
                    "-" +
                    dayList[i].month +
                    "-" +
                    dayList[i].day
                  );
                  options.labels.push(dayString);
                }

                xMonthArray.push(dayList[i].energy_wh);
              }
            } else {
              if (u == 0) {
                let dayString = String(
                  thisYear + "-" + thisMonth + "-" + i
                );
                options.labels.push(dayString);
              }
            }
          }
        });
      }
    });
  }
}
```

Student	Sam De Smedt - CMSDEV	
Datum	Taak	Tijd
1/07/2018	concept uitwerken	5,00
2/07/2018	concept uitwerken	3,00
3/07/2018	Laravel + Vue.js setup	3,00
4/07/2018	Laravel + Vue.js setup	2,50
5/07/2018	Laravel migrations, seeders, database, models, user auth,....	30,00
	Laravel migrations, seeders, database, models, user auth,....	
	Laravel migrations, seeders, database, models, user auth,....	
	Laravel migrations, seeders, database, models, user auth,....	
6/07/2018	Laravel migrations, seeders, database, models, user auth,....	4,00
	Laravel API + backoffice controllers, routing	10,50
	Laravel API + backoffice controllers, routing	7,00
	Laravel API + backoffice controllers, routing	5,00
	Laravel API + backoffice controllers, routing	9,00
	Laravel API + backoffice controllers, routing	6,50
8/07/2018	TP_LINK api research (API smart plugs)	7,75
9/07/2018	TP_LINK api research (API smart plugs)	9,00
	TP_LINK api research (API smart plugs)	
10/07/2018	Setup services.js (Outlet.js)	8,00
	Setup services.js	
	Concept visuals	
11/07/2018	Setup view: DASHBOARD	5,00
12/07/2018	Vue.js -> Tailwind integration, Setup view: DASHB	5,00
13/07/2018	Vue.js -> Tailwind integration	5,00
14/07/2018	Vue.js views - DASHB: CRUD, php controllers edit category relation	8,00
	Vue.js views - DASHB: CRUD, php controllers edit category relation	9,00
	Vue.js views - DASHB: CRUD, php controllers edit	5,00
18/07/2018	Vue.js views - DASHB: CRUD, php controllers edit	8,50
19/07/2018	Vue.js views - DASHB: CRUD, php controllers edit	7,50
20/07/2018	Vue.js views / Write scrips/php to store name so user can customise	4,00
	Vue.js views / Write scrips/php to store name so user can customise	8,00
21/07/2018	Vue.js views / implementatie sql database, err when device is removed/added	9,00
22/07/2018	Vue.js views - DASHB: CRUD, error logging (outlet offline)	5,00
23/07/2018	Vue.js privileges, roles, etc	8,50
24/07/2018	Vue.js views - USERS: CRUD, php controllers edit	7,50
25/07/2018	Vue.js views - USERS: CRUD, php controllers edit	4,00
26/07/2018	Vue.js views - USERS: CRUD, php controllers edit	8,00
27/07/2018	Vue.js & Laravel auth, view login page, Laravel passp	9,00
28/07/2018	Vue.js & Laravel auth, view login page, Laravel passp	5,00
29/07/2018	Vue.js views - ACTIONS -> lay out	8,50
30/07/2018	Research sensors, Rasberry Pi & firebase realtime database	12,00
	Research sensors, Rasberry Pi & firebase realtime database	
	Research sensors, Rasberry Pi & firebase realtime database	
	PI: python try get data from sensor, push to firebase	
31/07/2018	PI: python try get data from sensor, push to firebase	5,00
1/08/2018	Research sensors try DHT11 => fail (gave wrong temp)	5,00
2/08/2018	Research sensors, try 18B20 => success!	7,50
	get/push to realtime DB from Vue.js	
3/08/2018	Functions in python	4,00
	Functions in python	
4/08/2018	Vue.js views ACTIONS, implemented actions from temp senso	8,50
	Vue.js views ACTIONS, implemented action from temp sensor	7,50
	Vue.js views ACTIONS, implemented actions from TP-LINK usage sensors	4,00
	Vue.js views ACTIONS, implemented actions from TP-LINK usage sensors	8,00
5/08/2018	Vue.js new idea: give devices category so can be controlled in ACTIONS	5,00
6/08/2018	DASH: give devices category: edit device edit-page (dropdown)	8,50
7/08/2018	LOCAL DATABASE: add 'category' table and many-to-many relation	3,00
8/08/2018	LOCAL DATABASE: alter 'devices' for new tables, rollback migr etc	4,00
9/08/2018	Vue.js views STATS, with data from emeter, layout plugin: apexcharts.js	18,00
10/08/2018	Vue.js views STATS, with data from emeter, layout plugin: apexcharts.js	
11/08/2018	Vue.js views STATS, with data from emeter, layout plugin: apexcharts.js	
12/08/2018	Vue.js views STATS, with data from emeter, live usage counter	
13/08/2018	Vue.js views STATS, with data from emeter, live usage counter	
14/08/2018	ACTIONS: error logging, features enable/disable sensor	
15/08/2018	ACTIONS: features enable/disable sensor, list devices per category	
16/08/2018	Werken aan app.xd, productiedossier	
	Werken aan app.xd, productiedossier	
		TOTAAL
		345,75

Student	Sam De Smedt - CMSDEV	
Datum	Taak	Tijd
1/07/2018	concept uitwerken	5,00
2/07/2018	concept uitwerken	3,00
3/07/2018	Laravel + Vue.js setup	3,00
4/07/2018	Laravel + Vue.js setup	2,50
5/07/2018	Laravel migrations, seeders, database, models, user auth,...	30,00
	Laravel migrations, seeders, database, models, user auth,...	
	Laravel migrations, seeders, database, models, user auth,...	
	Laravel migrations, seeders, database, models, user auth,...	
6/07/2018	Laravel migrations, seeders, database, models, user auth,...	4,00
	Laravel API + backoffice controllers, routing	10,50
	Laravel API + backoffice controllers, routing	7,00
	Laravel API + backoffice controllers, routing	5,00
	Laravel API + backoffice controllers, routing	9,00
	Laravel API + backoffice controllers, routing	6,50
8/07/2018	TP_LINK api research (API smart plugs)	7,75
9/07/2018	TP_LINK api research (API smart plugs)	9,00
	TP_LINK api research (API smart plugs)	
10/07/2018	Setup services.js (Outlet.js)	8,00
	Setup services.js	
	Concept visuals	
11/07/2018	Setup view: DASHBOARD	5,00
12/07/2018	Vue.js -> Tailwind intergration, Setup view: DASHB	5,00
13/07/2018	Vue.js -> Tailwind intergration	5,00
14/07/2018	Vue.js views - DASHB: CRUD, php controllers edit category relation	8,00
15/07/2018	Vue.js views - DASHB: CRUD, php controllers edit category relation	9,00
	Vue.js views - DASHB: CRUD, php controllers edit	5,00
18/07/2018	Vue.js views - DASHB: CRUD, php controllers edit	8,50
19/07/2018	Vue.js views - DASHB: CRUD, php controllers edit	7,50
20/07/2018	Vue.js views / Write scrips/php to store name so user can customise	4,00
	Vue.js views / Write scrips/php to store name so user can customise	8,00
21/07/2018	Vue.js views / implementatie sql database, err when device is removed/added	9,00
22/07/2018	Vue.js views - DASHB: CRUD, error logging (outlet offline)	5,00
23/07/2018	Vue.js privileges, roles, etc	8,50
24/07/2018	Vue.js views - USERS: CRUD, php controllers edit	7,50
25/07/2018	Vue.js views - USERS: CRUD, php controllers edit	4,00
26/07/2018	Vue.js views - USERS: CRUD, php controllers edit	8,00
27/07/2018	Vue.js & Laravel auth, view login page, Laravel passp	9,00
28/07/2018	Vue.js & Laravel auth, view login page, Laravel passp	5,00
29/07/2018	Vue.js views - ACTIONS -> lay out	8,50

20/07/2018	Vue.js views / Write scrips/php to store name so user can customise	4,00
	Vue.js views / Write scrips/php to store name so user can customise	8,00
21/07/2018	Vue.js views / implementatie sql database, err when device is removed/added	9,00
22/07/2018	Vue.js views - DASHB: CRUD, error logging (outlet offline)	5,00
23/07/2018	Vue.js privileges, roles, etc	8,50
24/07/2018	Vue.js views - USERS: CRUD, php controllers edit	7,50
25/07/2018	Vue.js views - USERS: CRUD, php controllers edit	4,00
26/07/2018	Vue.js views - USERS: CRUD, php controllers edit	8,00
27/07/2018	Vue.js & Laravel auth, view login page, Laravel passp	9,00
28/07/2018	Vue.js & Laravel auth, view login page, Laravel passp	5,00
29/07/2018	Vue.js views - ACTIONS -> lay out	8,50
30/07/2018	Research sensors, Raspberry Pi & firebase realtime database	12,00
	Research sensors, Raspberry Pi & firebase realtime database	
	Research sensors, Raspberry Pi & firebase realtime database	
	PI: python try get data from sensor, push to firebase	
31/07/2018	PI: python try get data from sensor, push to firebase	5,00
1/08/2018	Research sensors try DHT11 => fail (gave wrong temp)	5,00
2/08/2018	Research sensors, try 18B20 => success!	7,50
	get/push to realtime DB from Vue.js	
3/08/2018	Functions in python	4,00
	Functions in python	
4/08/2018	Vue.js views ACTIONS, implemented actions from temp senso	8,50
	Vue.js views ACTIONS, implemented action from temp sensor	7,50
	Vue.js views ACTIONS, implemented actions from TP-LINK usage sensors	4,00
	Vue.js views ACTIONS, implemented actions from TP-LINK usage sensors	8,00
5/08/2018	Vue.js new idea: give devices category so can be controlled in ACTIONS	5,00
6/08/2018	DASH: give devices category: edit device edit-page (dropdown)	8,50
7/08/2018	LOCAL DATABASE: add 'category' table and many-to-many relation	3,00
8/08/2018	LOCAL DATABASE: alter 'devices' for new tables, rollback migr etc	4,00
9/08/2018	Vue.js views STATS, with data from emeter, layout plugin: apexcharts.js	18,00
10/08/2018	Vue.js views STATS, with data from emeter, layout plugin: apexcharts.js	
11/08/2018	Vue.js views STATS, with data from emeter, layout plugin: apexcharts.js	
12/08/2018	Vue.js views STATS, with data from emeter, live usage counter	
13/08/2018	Vue.js views STATS, with data from emeter, live usage counter	
14/08/2018	ACTIONS: error logging, features enable/disable sensor	9,00
15/08/2018	ACTIONS: features enable/disable sensor, list devices per category	5,00
16/08/2018	Werken aan app.xd, productiedossier	
	Werken aan app.xd, productiedossier	
		TOTAAL
		345,75