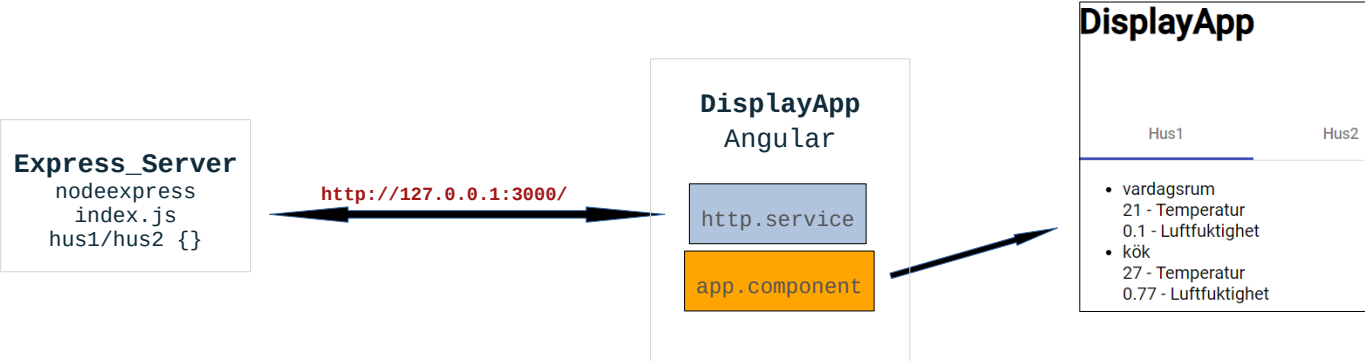# REST_Project overview

This project is set up to run on localhost. It's demonstrating a server application with Node.js with the Express framework, RESTful API, using Angular as front-end. Two objects, Hus1/Hus2, are fetched from the Express_server into DisplayApp via http service; then displayd in browser.

Express_Server
nodeexpress
index.js
hus1/hus2 {}

http://127.0.0.1:3000/

DisplayApp
Angular

http.service

app.component

DisplayApp

Hus1	Hus2

- vardagsrum
  21 - Temperatur
  0.1 - Luftfuktighet
- kök
  27 - Temperatur
  0.77 - Luftfuktighet

SW flow

Express_Server
index.js

```js
const hus1 = {
    rooms: [{
        name: 'vardagsrum',
        temperature: '21',
        humidity: '0.1'
    },{
        name: 'kök',
        temperature: '27',
        humidity: '0.77 '
    } ]};


homes.get('/homes/hus1/rooms', (req, res) => {
    const  hus1AllRooms = hus1.rooms;
    res.send(hus1AllRooms);  });
```

DisplayApp

http.service.ts

```ts
export interface Hus {
name: string;
temperature: number;
humidity: number;}


constructor(private httpClient: HttpClient) {}

getHus1(): Observable<Hus[]> {
return this.httpClient.get<Hus[]>(this.hus1Url); }


private hus1Url =
'http://127.0.0.1:3000/homes/hus1/rooms';


export class HttpServiceComponent {
@Injectable({ providedIn: 'root' })
```

app.component.ts

```ts
constructor(private httpService: HttpServiceComponent)
hus1Observable$ = this.httpService.getHus1();
```

app.component.html

```html
<mat-tab label="Hus1">
<ul *ngIf="hus1Observable$ | async as hus1Obj
                        else noDataHus1">
    <li *ngFor="let keyVal of hus1Obj">
        {{ keyVal.name }}
        {{ keyVal.temperature }}
        {{ keyVal.humidity }} </li></ul>


  <ng-template #noDataHus1>
     No Data Available for Hus1
  </ng-template>
 </mat-tab>
```

DisplayApp

| Hus1 | Hus2 |

- vardagsrum
  21 - Temperatur
  0.1 - Luftfuktighet
- kök
  27 - Temperatur
  0.77 - Luftfuktighet