

on \$
on \$
on \$

if
if
if

serve

S

S

S

☺ open browser

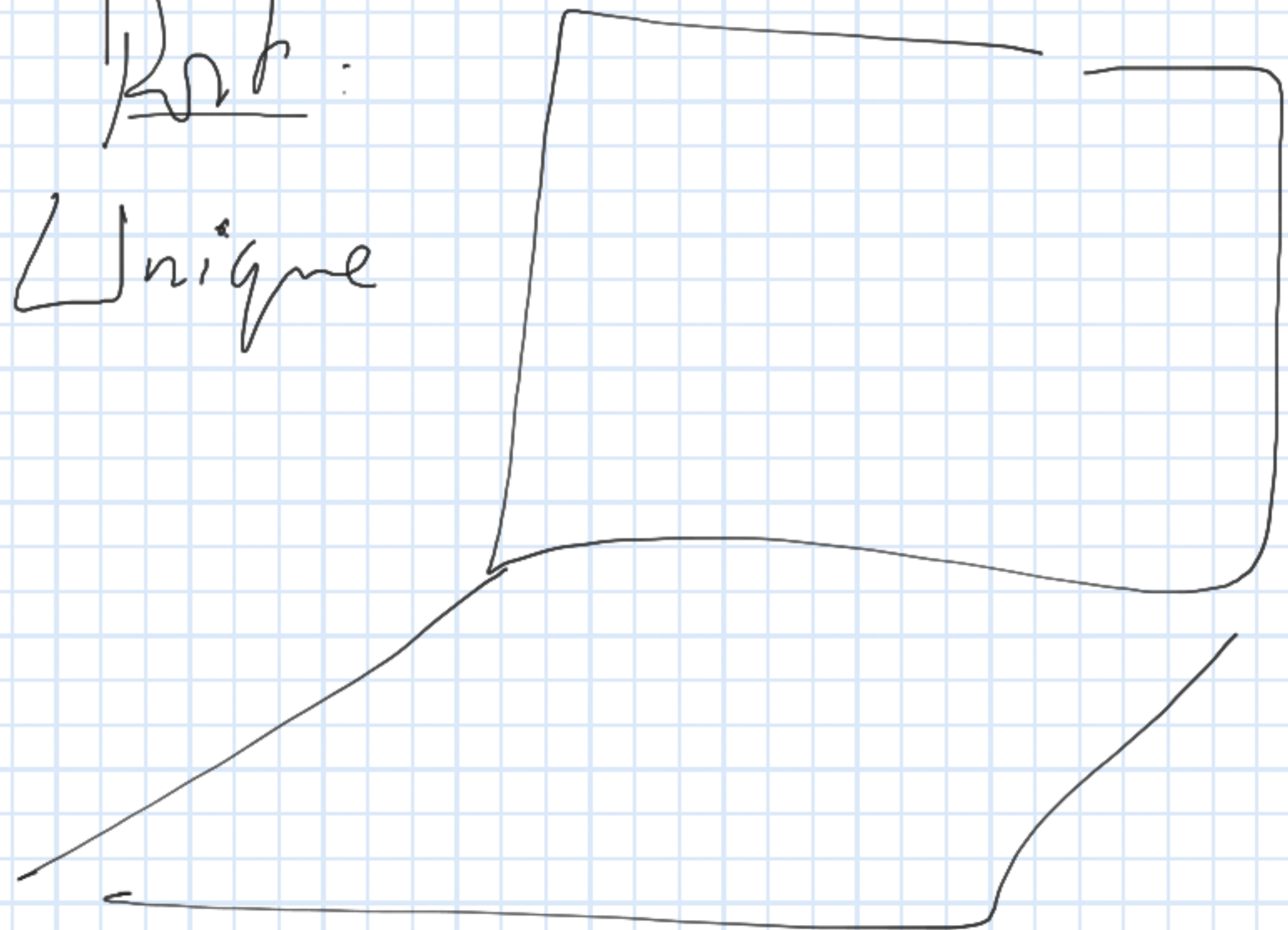
--port = 4205

☺

option
flag

http://
localhost:
4200

Root:
Unique



Processus

--- compiled Successfully

~~\$~~ ← prompt

Bash

\$ if s -o | \$

index.html

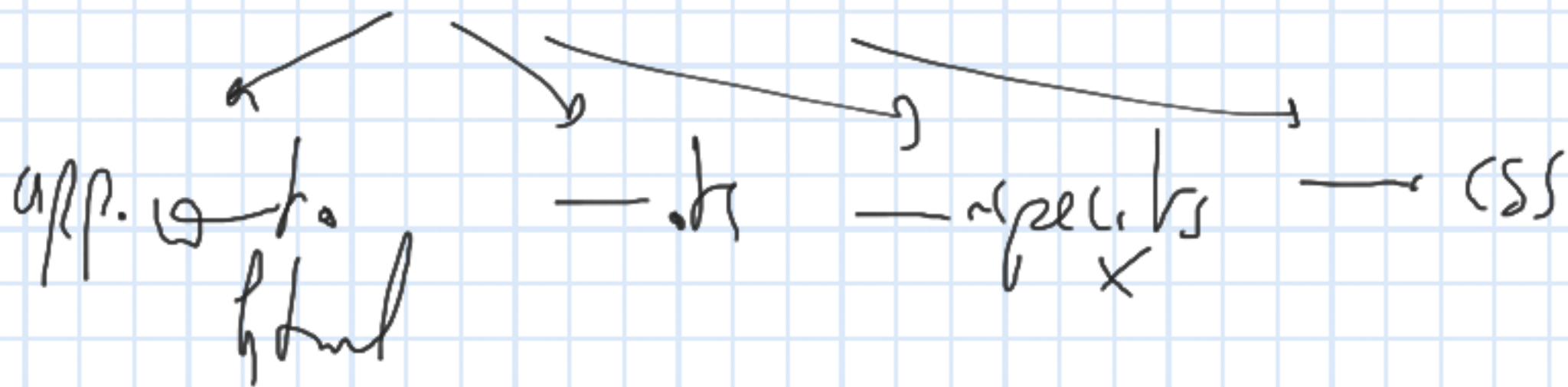
S/A

↓
<app-root>

selector

du Comp
Principal

AppComp



index

<app-root>

app-w — h.html

<app-header>

<router-outlet>

<app-footer>

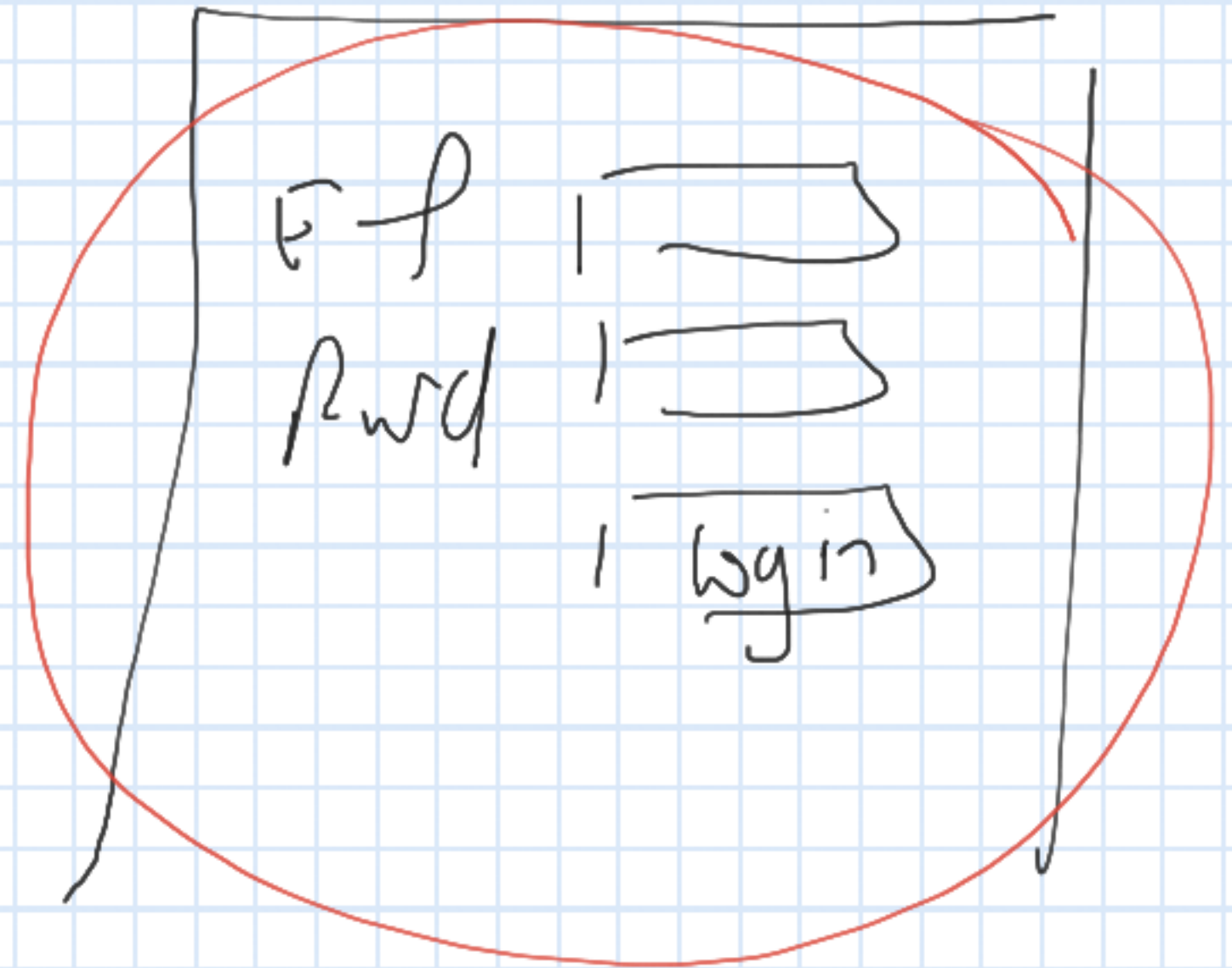
</app-header>

</router-outlet>

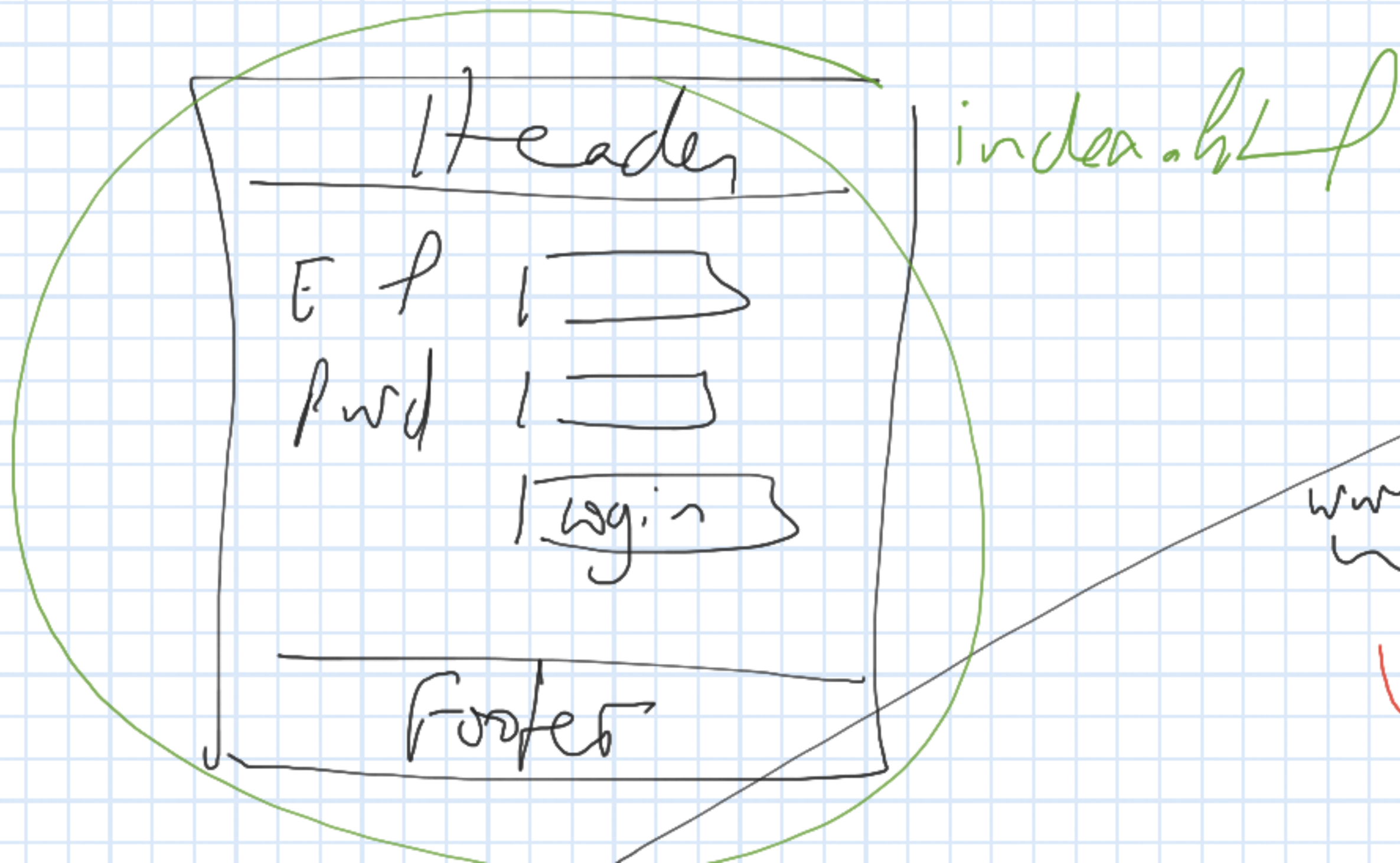
</app-footer>

AD

La libe Angular q: permet d'injecter
le contenu d' — Component selon
le path : index.html



http://localhost:4200 / Sign in
Base URL path



www.facebook.com
Proj. 6

app-routing.module.ts : fichiers de
configurat° des
paths

{ Path: 'signin', component: NomCont }

Component : bloc réutilisable d'une
page web ayant une partie graphique
et une partie logique (.ts) (-.html, .css)

Comp: — classe (—.js) ayant —
partie Graphique (—.html + .css)

{ attr: val, attr1: val1, ... } : objet
JSON

JavaScript Object Notation

- ensemble d'attributs, et de valeurs
- format de données
- l' d'échange de données.

router-outlet



app-routing.
module.ts

const

{

{

routes: Routes = [

},

}

];

Array
Table

export Class

A {

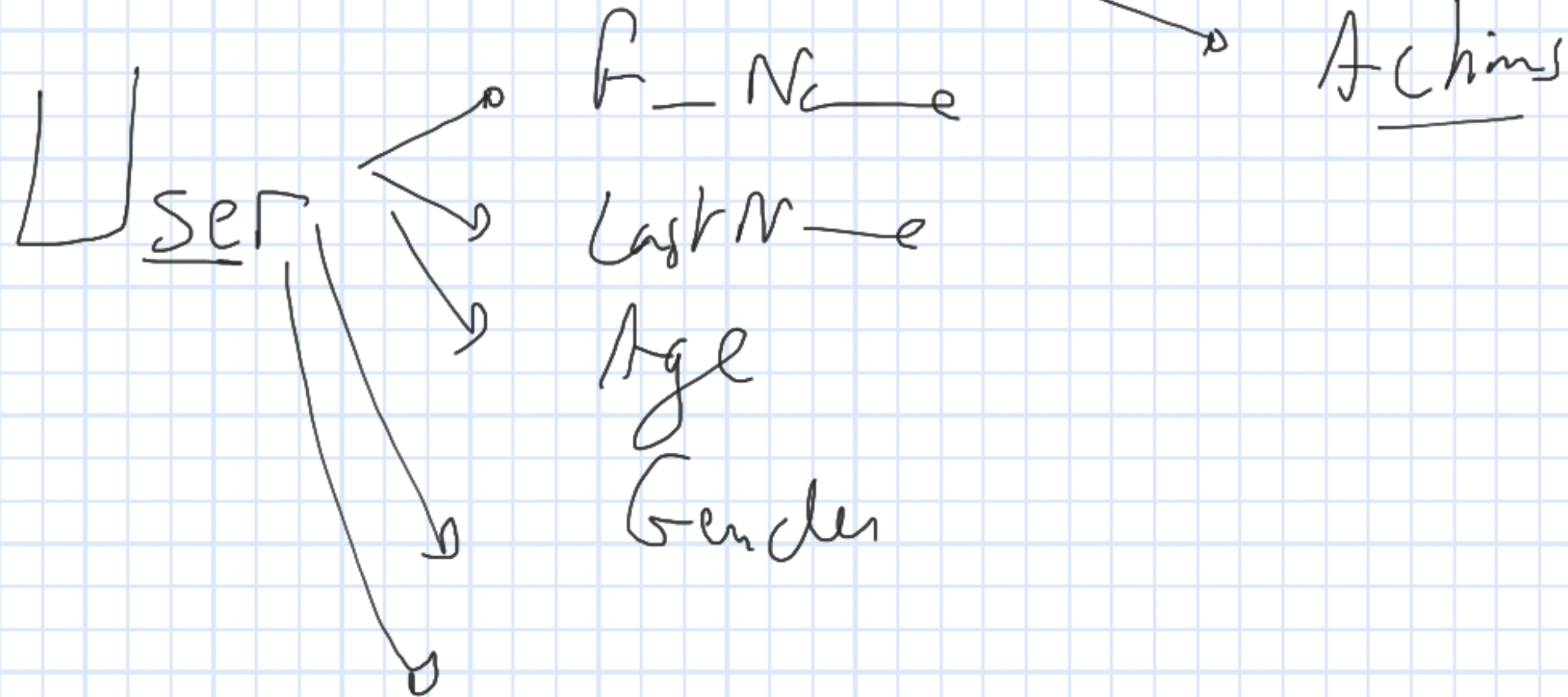
attrib / variables
globals

X: number = 10;
Y: string = 'hello';
Z: boolean = 'true';

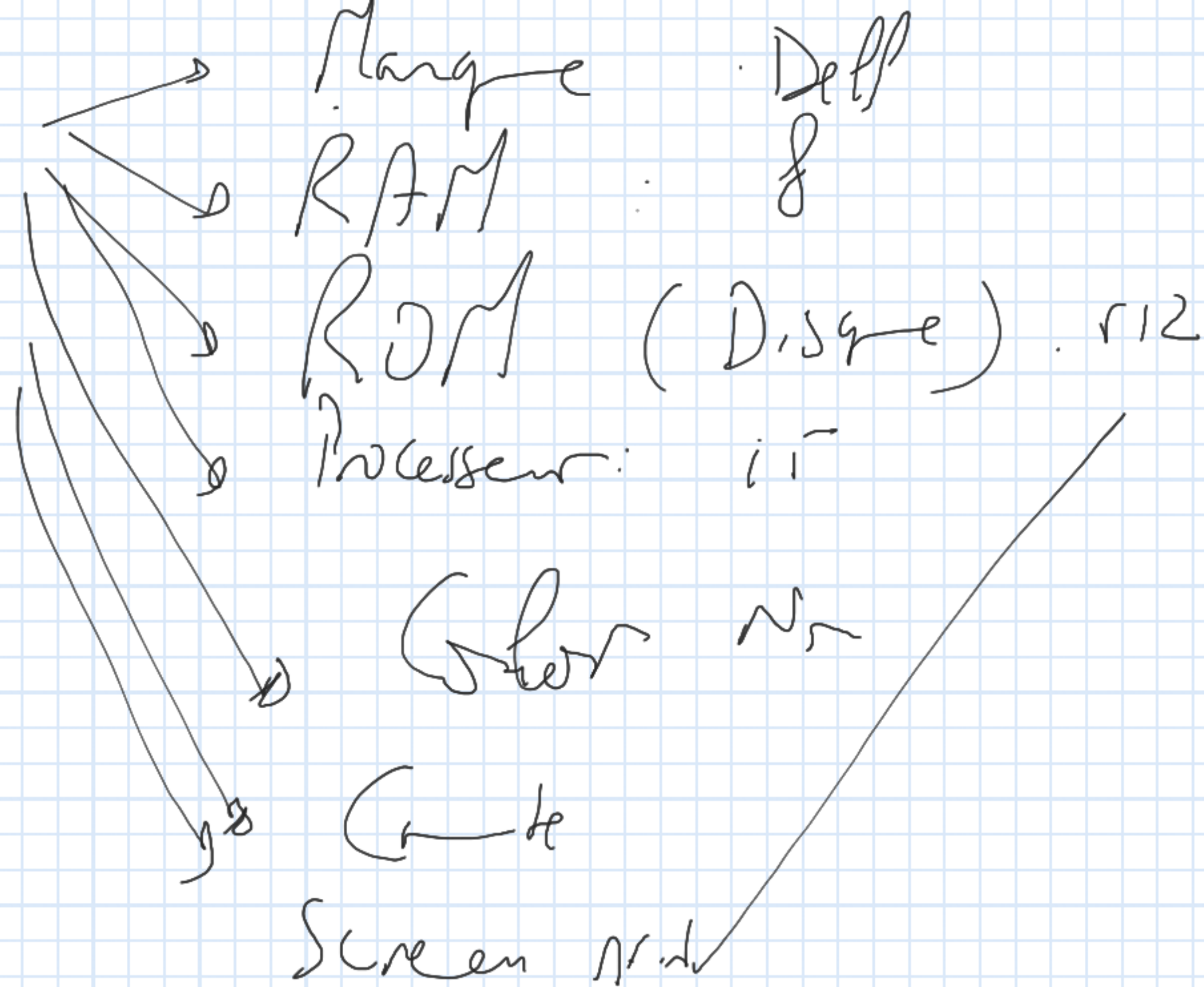
Methods

← {
 calc (a, b) {
 rel a + b;
 }
 sum (a, b, c) { rel a + b + c; }
}

une classe : une modélisation d'un objet
de la vie réelle ayant des attributs
et des méthodes collectifs



PC



On peut créer des instances (objet)
d'une classe

exptk class N-Class
Pour q' puisse // attributs
l'importer // méthodes

}

export

class A {

~~x~~ number = 12;

✓ y: boolean = true;

attributes

this: X

methods

calc(a, b) {

let ~~x~~ = 12;

return a + b * ~~x~~;

}

so (a, b, c) {

let z = 9;

return a + b + c - z + this.X;

}

}

Générer un \hookrightarrow :
\$ Se pointer ds le dossier comp

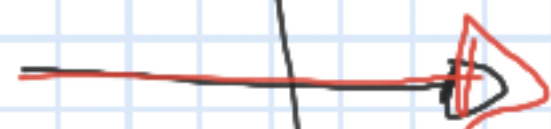
\$ \approx g c N \hookrightarrow match
add-match
generate \hookrightarrow

Data Binding : liaison entre la partie
Graphique et la partie logique

— .ts

x : number = 10;

calcul(2, 8) {
 return a + b;
}



SLF

{ { } } x { { } } 10
Interpolation
{ { calcul(2, 8) } }

x = "assets/images/a.jpg";

↳ attribute

y = "w";

display(
 alert(this.y + " matches");
)

—obs

extension

—obs

Property Binding

— o ds

```
displayValue() {  
  return 'btn btn-success';  
}
```

→

— o hhl

```
<button [class]=displayValue()  
  class='btn btn-success'
```