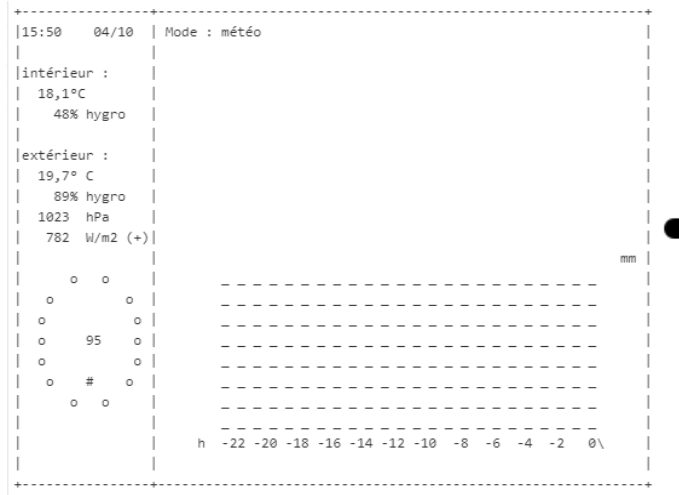




# Compléter un projet en cours

Ajouter des fonctionnalités en respectant le modèle graphique

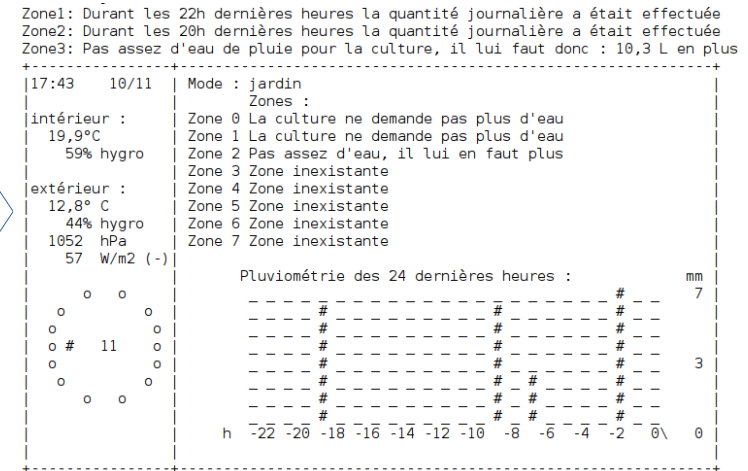
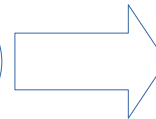
S'appropriier un code inconnu rapidement



-État initial



Cahier Des  
Charges

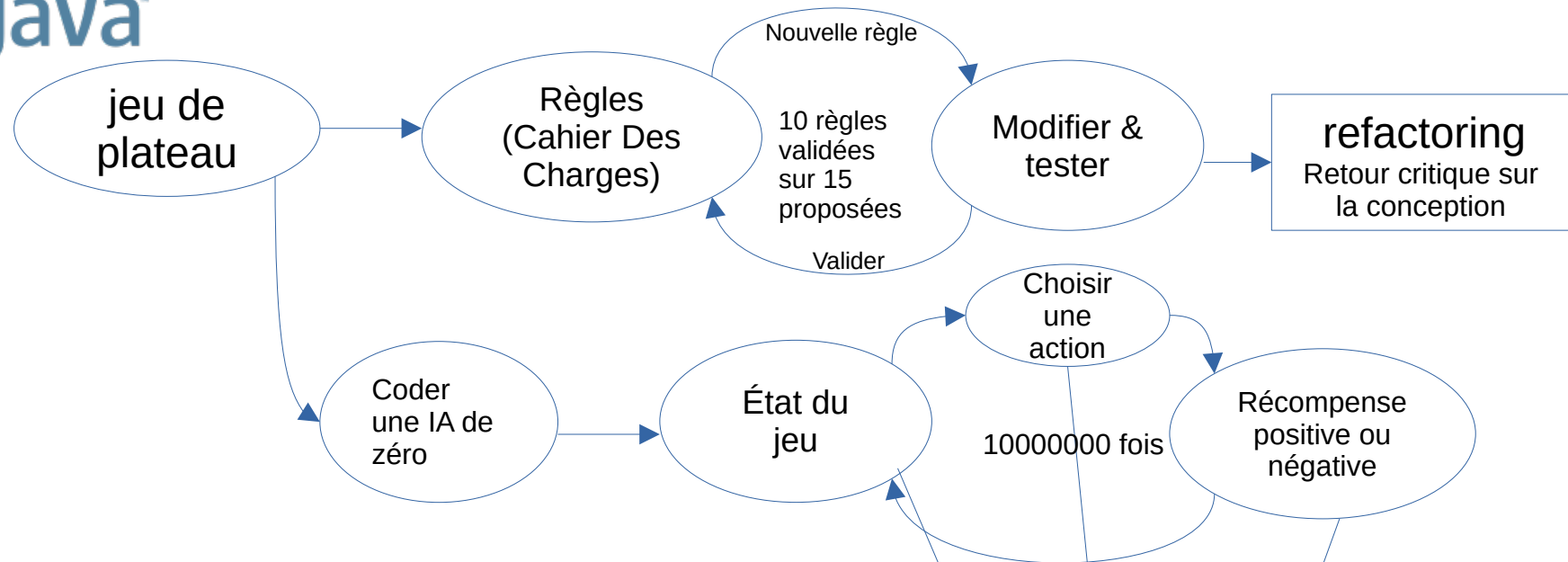


-État final



# Prendre des risques

Savoir préférer la qualité à la quantité



Q-learning

$$Q[s, a] := (1 - \alpha)Q[s, a] + \alpha \left( r + \gamma \max_{a'} Q[s', a'] \right)$$

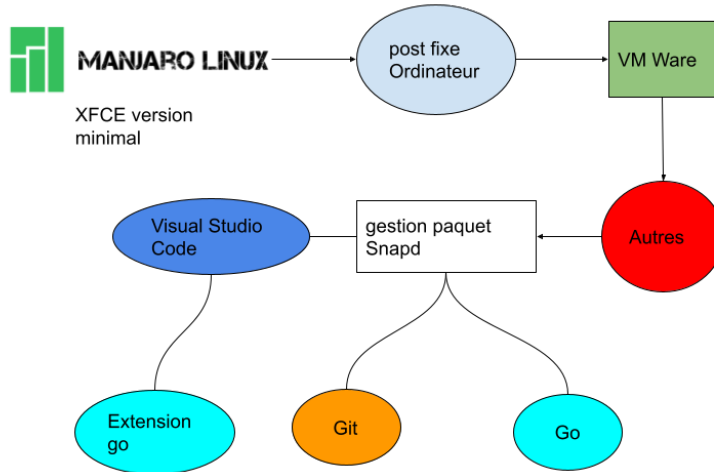
Code état	action jouer	résultat	score IA	score Adversaire
0	10	10	1	0
2	40	-10	2	3
4	20	0	4	4

# Installation d'une machine virtuelle

Respect des besoins du client & Anticipation

Création d'un livrable détailler

## Attendus et résultats



Utilisateurs : développeur et administrateur

```
[alice@manjaro etc]$ sudo whoami
root

[bob@manjaro ~]$
=>whoami
bob
```

Linux léger

Usage: 2.7 GiB of 3.8 GiB free (30% used)

Golang et Vscope

```
home > bob > Desktop > test.go
1 package main
2
3 import "fmt"
4
5 func main() {
6     fmt.Println("hello world")
7 }

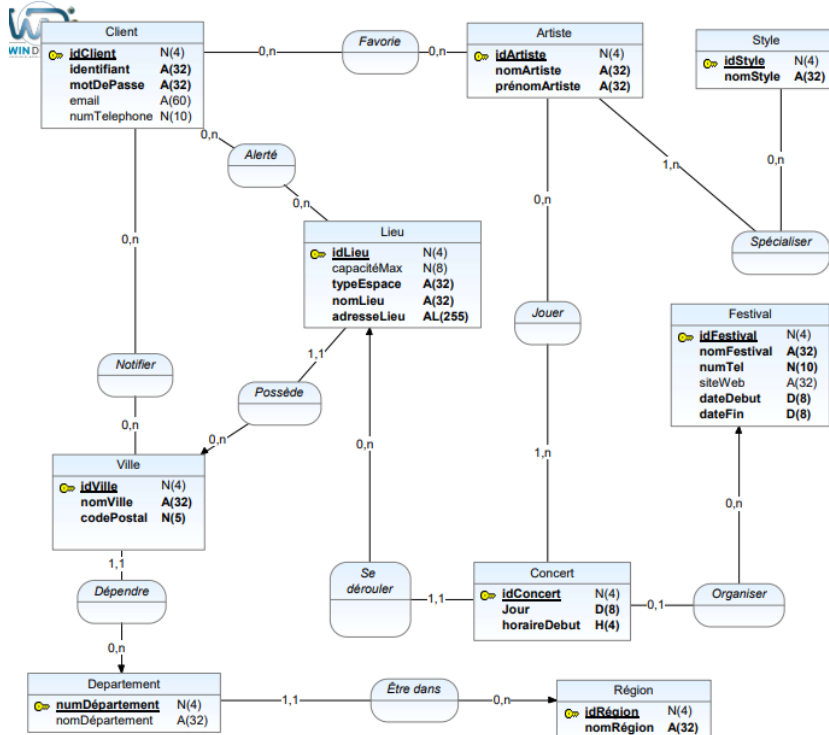
[Desktop]> bash
[bob@manjaro Desktop]$
* History restored

[bob@manjaro Desktop(master #status = new)]$
=>go run test.go
hello world
[bob@manjaro Desktop(master #status = new)]$
=>[]
```

# Création d'une base de données

Respect des besoins du client & Anticipation

Établir une jeu de données pertinent



# Création d'un site web

Créer et respecter la maquette du site

Gestion de l'UX/UI design, responsive

