**Solid:**

Responsabilité unique :

* On a divisé nos classes en sorte qu’ils ont un rôle uniquement :
* Class lirefichier – >Fait la lecture de notre string
* Class Instruction -> Permet de sauvegarder une instruction
* Class Territoire -> Permet de faire la capture d’un territoire
* Class Pass -> s’occupe de l’instruction pass
* Class Jeton -> s’occupe uniquement des informations en lien avec un jeton
* Class JeuGo ->son rôle est de traiter la partie pour l’afficher

Ouvert/Fermé:

* Pour accéder à nos méthodes ,il faut nécessairement utilisé des getters .
* Il est facile d’avoir de l’extensibilité dans notre code :
  + Par exemple : Jeton et Pass sont des sous- types de TypeInstruction .Dans le futur , il sera alors possible d’avoir d’autres types d’instructions .
  + Par exemple :Position est defini par rangee et colonne ,de cette facon position ,il est beaucoup plus facile dajouter des nouveaux types de positions(Position fait la validation de sa rangee et sa colonne pour verifier si ils existent)

Substitution de Liskov :

* Jeton et Pass hérite de typeInstruction .On peut remplacer typeInstruction par nimporte quelle sous type sans briser le code . On peut utiliser les méthodes de types instructions pour tous les enfants qui vont l’hériter sans que le code les brise

Ségrégation des interfaces :

* Les différentes méthodes dans jeu Go ont était distribuée dans différente interfaces qui peuvent êtres utilisées indépendamment les unes des autres .(Interface Mise a jour jeton et récupération Jeton)

Injection de dépendances :

-Dans la classe affichageJeu implémente l’interface affichage .Jeugo instancie l’interface affichage dans son constructeur . Dans la méthode affichage dans jeuGo font appel aux méthodes de l’interface affichage et dans scratsh on met en paramètre l’interface affichage.

**Grasp:**

Patron #1 Spécialiste de l’information

Instruction passe par position qui par la suite passe par JeuGo pour POUR AVOIR DANS UN TABLEAU

- Dans la classe JeuGo a une reference a toutes les informations instructions du jeu.JeuGo a la responsabilité de faire la gestion du jeu a partir des instructions.

### Patron#2  Créateur

Jeugo est composé d’instruction et la responsabilité de créer une instruction revient a jeugo.

Instruction est composé de jeton .La responsabilité de créer Jeton est d’instruction.

Deplus, comme jeugo contient toute l’information du jeu il permet de créer les territoires

Patron#3

Contrôleur

La class principal JeuGo est chargée de la responsabilité de contrôler les entrées provenant de LireFichier et reliée ses informations vers les objets interne du jeu.