**Plan de travail : Numéro 3**

**Cas coup de départ ( vérifier si on déplace une pièce de même couleur que la couleur contrôle ) :**

**if ( clique sur intersection occupée & iconeTampon vaut null )**

* **initialiser la position de départ**
* **prendre la pièce et l'assigner à pieceTampon**
* **prendre l'icône et l'assigner à iconeTampon**
* **enlever l’icône de son JLabel d'origine, ne pas enlever la pièce, pourquoi ? car on doit appeler les methode qu’on a codees (estvalide, cheminPossible) avatn de déplacer officeillement la piece**

**Cas coup d'arrivée ( sans capture de pièces ) :**

**else if ( clique sur intersection vide & pieceTampon n’est pas vide )**

* **initialiser position d'arrivée**
* **vérifier estValide**
* **vérifier cheminPossible**
* **vérifier si les Rois ne sont pas face-à-face**
* **si tout fonctionne :**
  + **enlever la pièce de la case de départ, comment ? setPiece(null)**
  + **mettre la piece tampon sur la case d'arrivée et vider le tampon, comment ?**
  + **placer l'iconeTampon sur le JLabel d'arrivée et l'enlever du tampon**
  + **donner le contrôle à l'autre Joueur et l'indiquer(bonus)**

**autre cas d'arrivée ( avec capture de pièce ) :**

**else if ( clique sur intersection occupée & iconeTampon n’est pas null )**

**idem au cas précédent, cependant il faut placer l’icône de la pièce capturée dans les JPanels appropriés(bonus)**

**penser à ne pas permettre auto-capture de pièce !**