**Ajout : Fonctionnalités.**

**FP FT FC**

**-Jeu**

-Joueur contre joueur

-Paramétrage de partie

-Couleurs des pions

-Nombre de couleur

-Nombre d’essais

-…

-Choix de la combinaison secrète.

-Validation de la combinaison, sans perte de la solution ou des propositions précédentes.

-Switch entre les joueurs (propositions/indices)

- Voir à délégué la production d’indices a la machine.

-Comptage des points + voir Scoring BDD (si rallonge temps).

-Joueur contre IA (IA propose, joueur découvre) -Même fonctionnalité avec combinaison aléatoire en proposition.

Interface graphique intégrée et fonctionnelle

|  |  |  |  |
| --- | --- | --- | --- |
| Fonction | Importance | Critères d’appréciations | Niveau d’appréciation |
| FP : |  |  |  |
| **FP1** : Paramétrage partie. | 4 | -Choix de la couleur des pions.  -Nombres d’essais.  -Nombre de pions de la combinaison secrète.  -Possibilité de réutiliser une même couleur. | - Minimum 6 couleurs,  maximum 8 couleurs.  - Minimum 10 d’essais et 15 max.  - Minimum 4 pions dans la combinaison secrète, Max 6 couleurs.  -La combinaison secrète peut avoir ou non plusieurs fois la même couleur. |
| **FP2** : Création de combinaison secrète. | 5 | -L’utilisateur doit choisir le nombre de pions définie préalablement.  -L’utilisateur a le choix dans les couleurs disponibles. | -Le nombre de pions dans les combinaisons est égale au nombre de pions choisie préalablement.  -Seuls les couleurs disponibles sont utilisés. |
| **FP3** : Proposition et validation de la solution. | 5 | -L’utilisateur doit pouvoir construire une solution ayant le bon nombre de pions et utilisant les couleurs associés à la partie.  -Le joueur doit pouvoir soumettre sa combinaison pour la comparer avec la combinaison secrète. | -La solution doit être en format comparable à la combinaison secrète.  -Le joueur doit connaître les pions de bonnes couleurs bien placés.  -Le joueur doit connaître les pions de bonnes couleurs mal placés. |
| FC : |  |  |  |
| **FC1** : Fenêtre graphique de partie. | 4 | -Une fenêtre graphique s’affiche pour modéliser la partie en cours.  -La combinaison secrète doit pouvoir être occultée.  -La proposition doit être visible et modifiable à partir de cette fenêtre.  -Le joueur doit pouvoir savoir combien d’essai il a effectué. | -La fenêtre doit afficher les solutions.  -la combinaison secrète n’est pas visible.  -La composition de la proposition ce fait via l’interface graphique.  -Le nombre d’essai est affiché (explicitement ou implicitement). |
| **FC2** : Fenêtre graphique du menu. | 3 | -Une fenêtre graphique s’affiche pour pouvoir lancer une nouvelle partie ou voir les crédits. | -La fenêtre possède un bouton pour lancer le paramétrage de partie.  -La fenêtre possède un deuxième bouton pour lancer la fenêtre de crédit. |
| **FC3** : Fenêtre graphique de paramétrage de partie. | 3.5 | -Une fenêtre graphique doit s’afficher pour que l’utilisateur puisse modifier les paramètres de la partie qu’il va lancer. | -Un objet graphique permet le choix du nombre de couleurs dans les combinaisons.  -Un objet graphique permet le choix du nombre de pions dans les combinaisons.  -Un objet graphique permet le choix du nombre d’essais disponible aux joueurs.  -un objet graphique permet l’autorisation du choix multiple de couleurs dans une combinaison. |
| **FC4** : Fenêtre graphique des crédits. | 2 | -Une fenêtre graphique s’affiche pour que l’utilisateur affiche les crédits du programme. | -Un texte affiche tous … |
| **FT :** |  |  |  |
| **FT1** : Comptage du score. | 3 | -Le score doit être calculé. | -Le score calculé doit correcte à 100%.  -Le score est calculé à partir du nombre d’essais effectués. |
| **FT2** : Switch entre les joueurs. | 5 | -Le programme doit pouvoir gérer le tour des joueurs. | -Le programme ne commet aucune erreur sur le tour de jeux.  -Le 1er joueur choisit une combinaison secrète.  -Puis le 2ème joueur émet une proposition jusqu’à ce que le nombre essais soit épuisé ou que la proposition soit valide.  -Le 1er joueur compare à chaque proposition du 2ème joueur. ( -> voir pour proposer comparaison automatique). |

# • 3. Cadre de réponse

**Diagramme de classe général du mastermind.**

*Diagramme de classes métiers : à base de classes métiers (des concepts indépendants de la programmation, avec des attributs mais sans méthodes, ainsi que des relations entre les concepts sur les agissements des classes métiers les unes sur les autres)*

*Synthèse des jalons : tables synthétiques des jalons du projet.*

## • 3.1 Pour chaque fonction.

### • 3.1.1 Solution proposée.

*Tâches à réaliser pour la solution -> WBS*

*Diagramme de séquence basée sur le diagramme de classe métier et sur les scénarios des cas d’utilisation (cf. section 2)*

*Jalon : date à laquelle la fonction doit être livrée.*

### •3.1.2 Niveau atteint pour chaque critère d’appréciation de cette fonction et modalités de contrôle

### • 3.1.3 Part du prix attribué à chaque fonction

• 3.2 Pour l’ensemble du produit

### • 3.2.1 Prix de la réalisation de la version de base

### • 3.2.2 Options et variantes proposées non retenues au cahier des charges

### • 3.2.3 Mesures prises pour respecter les contraintes et leurs conséquences économiques

### • 3.2.4 Outils d’installation, de maintenance … à prévoir

### • 3.2.5 Décomposition en modules, sous-ensembles

### • 3.2.6 Prévisions de fiabilité