Cahier des charges

# Révisions

|  |  |  |
| --- | --- | --- |
| **Date** | **Version** | **Commentaires** |
| **23/04/2015** | 1.0 | Rédaction initiale |
|  |  |  |
|  |  |  |
|  |  |  |

# Description du projet

Contexte, description des fonctionnalités principales, etc

Jeu d’échec entre 2 joueurs avec les éléments suivants :

* Plateau
* Pièces joueur
  + Roi
  + Reine
  + Fou
  + Cheval
  + Tour
  + Pion

# Fonctionnalités principales

Version de base qui devra être fonctionnelle sous deux semaines

* déplacer une pièce selon les règles du jeu
* supprimer les pièces
* changer le joueur
* savoir si un roi est en échec

# Fonctionnalités avancées

Version améliorée de votre projet avec des fonctionnalités « avancées »

* coups possibles en surbrillance
* les droits au roque
* affichage du profil joueur avec score
* possibilité d’enregistrer une partie et de la rédemarrer

# Déroulement du projet

**1ère étape**

Création d’une partie

Positionnement des pièces

Définition aléatoire du premier joueur

Déplacement des pièces sans contraintes

**2ème étapes**

Déplacement des pièces selon les règles du jeu

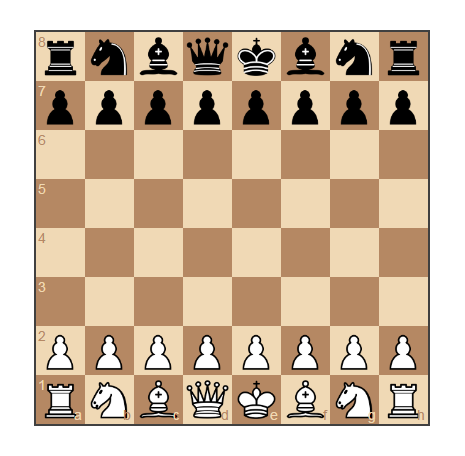
**3ème étapes  (règles)**

Définition des mouvements autorisés

Authentification des joueurs

Enregistrement de la partie

# Storyboard

PAGE D’ACCUEIL

Joueur Blanc

Joueur Noir

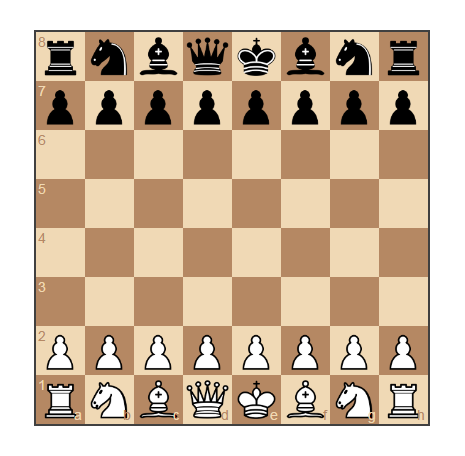
**Se connecter**

**S’enregistrer**

**Commencer une nouvelle partie**

**LOGO**

*Regles du Jeu / Credits*

VUE JOUEUR CONNECTÈ

Joueur Blanc

Joueur Noir

**BONNE CHANCE**

**Pseudo !**

*Regles du Jeu / Credits*

**Commencer une nouvelle partie**

**LOGO**

**VUES**

LOGIN

**LOGO**

**Pseudo : \_\_\_\_\_\_\_\_\_**

**Password : \_\_\_\_\_\_\_\_**

**Se connecter**

S’ENREGISTRER

**LOGO**

**Pseudo : \_\_\_\_\_\_\_\_\_**

**Email : \_\_\_\_\_\_\_\_\_\_\_**

**Password : \_\_\_\_\_\_\_\_**

**S’enregistrer**

CREDITS et REGLES DU JEU

**LOGO**

**<Texte>**

FIN DE PARTIE

**LOGO**

**YOU WON !!!**

**Commencer une nouvelle partie**

# Diagramme de classes

**Chessboard**

$id

$case (array x,y)

validatedMove ()

updateBoard()

killPiece()

isInside()

**Piece**

$id

$color

$position

move()

**King**

move()

**Queen**

move()

**Knight**

move()

**Rook**

move()

**Pawn**

move()

**User**

$email

$pseudo

$password

**Player**

$color

$turn (Boolean)

play ()

**Bishop**

move()

**Listes de Taches**

« Sans contraintes »

* Configuration du projet Symfony
  + Création Entités et Contrôleurs sur Symfony
* PHP : Initialisation du plateau
  + Positionnement des pièces
* Création d’une partie
  + Affichage du plateau, Traitement des images
* Définition aléatoire du premier joueur
* Déplacement des pièces sans contraintes
* Mise à jour du plateau – Ajax/JQuery

« Avec contraintes »

* Déplacement des pièces selon les règles
* Suppression des pièces
* Savoir si un roi est en échec /fin de la partie

« Fonctions avancées »

* Inscription utilisateur
* Connexion utilisateur
* Affichage du score
* Surbrillance des cases autorisées pour le déplacement
* Enregistrement d’une partie