

Analyse de iSudoku

Projet de l'UE Ingénierie du Logiciel

Maude BELLAMY Antoine HOUSAIS Théo LEBOURG Jérôme
RAHAULT Fabricio SANTOLIN DA SILVA Simon TCHERNIA

Université Pierre et Marie Curie

10 décembre 2013

1 Phase de conception

- Diagramme de composant et interfaces requises/offertes

1 Phase de conception

- Diagramme de composant et interfaces requises/offertes
- Quelques diagrammes de séquence de niveau interaction inter-composant

1 Phase de conception

- Diagramme de composant et interfaces requises/offertes
- Quelques diagrammes de séquence de niveau interaction inter-composant
- Instanciation nominale des composants

1 Phase de conception

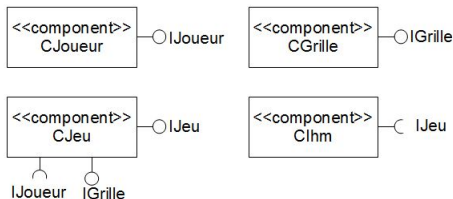
- Diagramme de composant et interfaces requises/offertes
- Quelques diagrammes de séquence de niveau interaction inter-composant
- Instanciation nominale des composants
- Diagrammes de classe de niveau conception détaillée

1 Phase de conception

- Diagramme de composant et interfaces requises/offertes
- Quelques diagrammes de séquence de niveau interaction inter-composant
- Instanciation nominale des composants
- Diagrammes de classe de niveau conception détaillée
- Tests d'intégration

2 Génération de code

Notre diagramme de composant et leurs interfaces requises/offerts



| <<interface>> IJoueur |
|---|
| + afficherStatistiques() : Image + getBonus() : int + getQl() : int + ajouterBonus() + ajouterDonnée(int temps, int difficultéGrille, int niveauAide, boolean partieGagnée) + publierRésultats(String login, String mdp) |

| <<interface>> IGrille |
|---|
| + getNuméro(int ligne, int colonne) : int + setNuméro(int ligne, int colonne, int val) |

| <<interface>> IJeu |
|---|
| + définirNiveauAide(int niveau) + nouvellePartie(IGrille g) + faireProposition(int ligne, int colonne, int val) : boolean + afficherPartie() : Image + grilleComplète() : boolean + validerGrille() + possibilitésCase(int colonne, int ligne) : int[*] |

Diagramme de séquence « Jouer une partie »

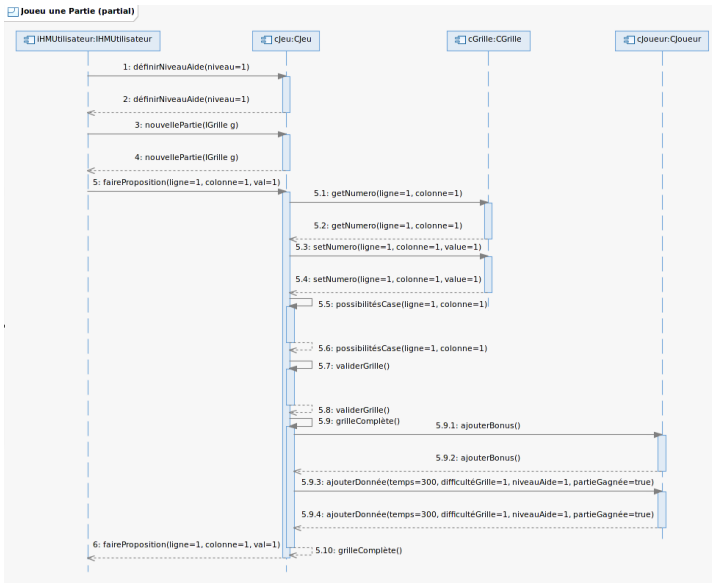


Diagramme de séquence « Affiche statistiques »

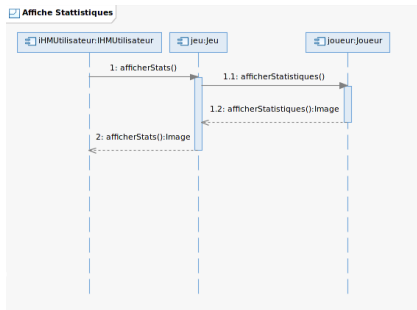
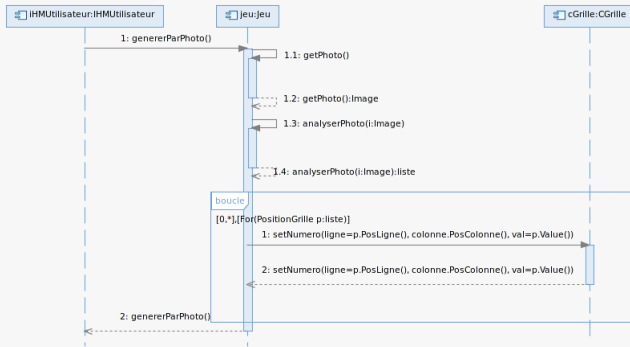


Diagramme de séquence « Générer une grille par photo »

Generer Par Photo



`getPhoto():Image` is activated after the user press the button to "Generer par Photo". The software built in into the phone will handle taking the picture and saving in the device. The device will return to the function an image that we pass it to the component "Jeu".

`analyserPhoto(Image i)` identify the grid in the image and get its values. It then returns a grid to be use in memory in the game.

Instanciation nominale des composants

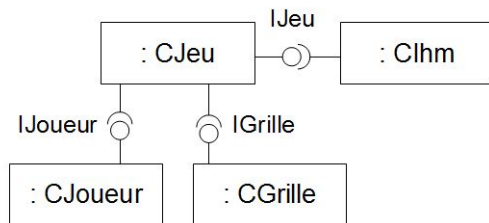


Diagramme de classe de niveau conception détaillée

CGrille

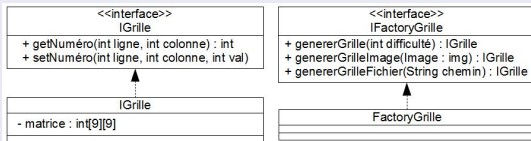
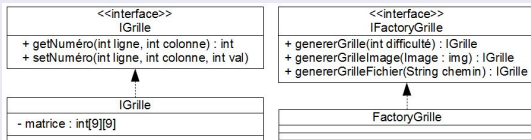


Diagramme de classe de niveau conception détaillée

CGrille



CJeu

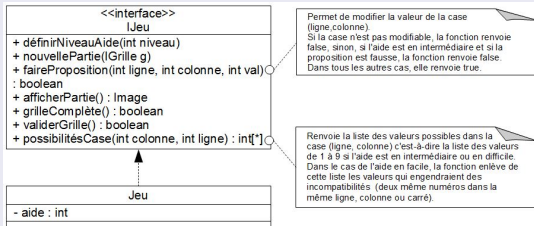
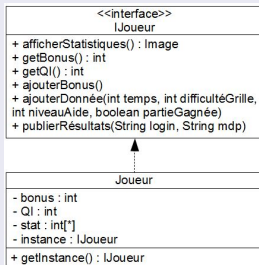


Diagramme de classe de niveau conception détaillée

CJoueur



Test d'intégration n° 1 du cas d'utilisation « Jouer une valeur en difficulté facile »

Titre

Terminer une partie en jouant une valeur correcte en niveau facile

Test d'intégration n° 1 du cas d'utilisation « Jouer une valeur en difficulté facile »

Titre

Terminer une partie en jouant une valeur correcte en niveau facile

Contexte

Une partie a été lancée en niveau « facile » (cf. Test n° x) et il reste une case à compléter

Scénario

- 1 L'utilisateur clique sur la case à remplir
- 2 Le système affiche la valeur possible
- 3 L'utilisateur entre la valeur correcte
- 4 Le système valide la grille et affiche les statistiques de la partie

Test d'intégration n° 1 du cas d'utilisation « Jouer une valeur en difficulté facile »

Titre

Terminer une partie en jouant une valeur correcte en niveau facile

Contexte

Une partie a été lancée en niveau « facile » (cf. Test n° x) et il reste une case à compléter

Scénario

- 1 L'utilisateur clique sur la case à remplir
- 2 Le système affiche la valeur possible
- 3 L'utilisateur entre la valeur correcte
- 4 Le système valide la grille et affiche les statistiques de la partie

Résultat attendu

Le partie est terminée.

Moyen de vérification

Visuelle : un message annonçant que la grille a été correctement complétée est affiché

Test d'intégration n° 2 du cas d'utilisation « Générer une grille via une photo »

Titre

Générer une grille de iSudoku en prenant une photo

Test d'intégration n° 2 du cas d'utilisation « Générer une grille via une photo »

Titre

Générer une grille de iSudoku en prenant une photo

Scénario

- 1 L'utilisateur clique sur le bouton « prendre une photo »
- 2 L'utilisateur prend une photo d'une grille de Sudoku
- 3 Le système affiche que la grille a bien été générée
- 4 Le système affiche la grille

Test d'intégration n° 2 du cas d'utilisation « Générer une grille via une photo »

Titre

Générer une grille de iSudoku en prenant une photo

Scénario

- 1 L'utilisateur clique sur le bouton « prendre une photo »
- 2 L'utilisateur prend une photo d'une grille de Sudoku
- 3 Le système affiche que la grille a bien été générée
- 4 Le système affiche la grille

Résultat attendu

La grille « numérisée » est identique à celle prise en photo.

Moyen de vérification

Visuelle : la grille apparaît sur l'application et est prête à être remplie

1 Phase de conception

1 Phase de conception

1 Phase de conception

1 Phase de conception

1 Phase de conception

2 Génération de code

Diagramme de classe

