



# Jeu des 7 différences

LOG2990 - Projet de logiciel  
d'application Web

**v18.3**

## Table des matières

<b>Description générale du projet.....</b>	<b>3</b>
Nombre de joueurs .....	3
Modes de jeu .....	3
<b>Vues.....</b>	<b>4</b>
Vue initiale .....	4
Vue de sélection de partie .....	4
Commencer une partie.....	5
Vue d'attente .....	5
Vue de jeu .....	5
Chronomètre .....	5
Compteur de différences trouvées.....	5
Section de messages.....	6
Identification d'une différence .....	6
Erreur d'identification .....	6
Vue d'administration .....	7
Créer un jeu en mode <i>point de vue simple</i> .....	7
Créer un jeu en mode <i>points de vue multiples</i> .....	8
Supprimer et réinitialiser un jeu.....	8
<b>Description des messages .....</b>	<b>9</b>
<b>Gestion et déroulement des parties .....</b>	<b>10</b>
Jumelage .....	10
Logique de jumelage .....	10
Initialisation .....	11
Jeu.....	11
Fin de partie .....	11
Mise à jour des meilleurs temps.....	11
<b>Composition d'un jeu .....</b>	<b>12</b>
Définition d'une différence .....	13
Exécutable générateur d'images de différence.....	15
Création d'un jeu en point de vue simple.....	15
Création d'un jeu en points de vue multiples .....	16
Génération d'images .....	16
Exécutable générateur d'images .....	18

## Description générale du projet

Le projet consiste à développer une version web du jeu des 7 différences, aussi appelé le jeu des 7 erreurs. Le concept de ce jeu est très simple. Deux images sont présentées au joueur : une originale, et une modifiée. Entre les deux, il se trouve sept différences que le joueur doit identifier.

Historiquement, on trouve de ces petits jeux dans diverses publications imprimées. Le format se prête toutefois bien à une adaptation informatisée, qui par la même occasion permet d'ajouter d'intéressantes fonctionnalités.

Dans notre cas, on veut améliorer le jeu en permettant les parties à deux joueurs et en développant une nouvelle façon de jouer.

## Nombre de joueurs

Le site doit offrir deux possibilités. La partie *solo* qui se joue seule, et le *un contre un* où deux joueurs s'affrontent.

Le jeu *solo* représente la façon classique de jouer. C'est-à-dire que le joueur cherche les différences et la partie se termine lorsqu'il les a toutes trouvées.

En un contre un, le but n'est pas de trouver toutes les différences, mais plutôt d'en trouver un certain nombre selon le mode jeu. Le premier joueur à atteindre ce seuil est déclaré vainqueur.

## Modes de jeu

Tel que mentionné, le jeu des 7 différences se joue à l'aide d'une image originale et d'une modifiée. C'est ce que nous appellerons le mode « point de vue simple ».

Pour notre version web, nous ajoutons une nouvelle façon de jouer où, à la place de présenter une seule image originale il doit y en avoir deux. Ces images doivent être deux clichés d'une même scène, mais chacune prise d'un point de vue différent. Bien entendu, deux images modifiées aux points de vue correspondants à ceux des originales doivent aussi être présentes. Nous utiliserons l'expression « points de vue multiples » pour nommer ce mode.

## Vues

Voici les différentes vues du site, leur contenu et les interactions possibles avec elles.

### Vue initiale

Il s'agit de la page d'accueil du site présentant le nom du jeu, son logo et un champ permettant à l'utilisateur d'y entrer un nom de joueur. Le nom est obligatoire et doit être validé par le système. Lorsque le nom choisi est autorisé, le joueur est redirigé vers la vue de sélection de partie.

### Vue de sélection de partie

C'est ici que le joueur effectue un choix parmi les jeux proposés. Ceux-ci sont présentés sous forme de petites fiches ayant le contenu suivant :

- Le nom du jeu
- Une vignette de l'image d'origine
- La liste des trois meilleurs temps en *solo*
- La liste des trois meilleurs temps en *un contre un*
- Un bouton pour jouer en *solo*
- Un bouton pour joindre ou créer<sup>1</sup> une partie en *un contre un*

*Note : Pour les jeux à points de vue multiples où il y a deux images d'origine, affichez seulement la vignette pour une des deux images.*



Exemple d'une fiche de jeu

<sup>1</sup> L'inscription sur le bouton dépendra de s'il y a un joueur en attente de jumelage pour faire une partie. Si c'est le cas, l'inscription sera *joindre*. Sinon, ce sera *créer*. Voir la section *Jumelage* pour plus de détails.

Les fiches sont organisées en deux colonnes, une par mode de jeu. La colonne de gauche pour les jeux en *point de vue simple* et la droite pour les *points de vue multiples*. Lorsque le joueur fait défiler la page pour voir les autres items de la liste, les deux colonnes bougent en même temps. Autrement dit, on ne peut pas manipuler les colonnes individuellement.

#### Commencer une partie

Sur une fiche de jeu, le joueur clique sur le bouton *jouer* pour faire une partie *solo* ou sur le bouton *joindre* ou *créer* pour faire une partie en *un contre un*. Un clic sur *jouer* ou *joindre* redirige aussitôt le joueur vers la vue de jeu. Un clic sur *créer* le redirige plutôt vers la vue d'attente.

#### Vue d'attente

Cette vue affiche un message demandant au joueur de patienter jusqu'à ce qu'un adversaire se manifeste. Elle lui offre aussi la possibilité d'annuler sa demande de création de partie, auquel cas il est retourné à la vue de sélection.

#### Vue de jeu

La vue de jeu présente le nécessaire pour qu'une partie puisse être jouée. Elle doit contenir les images originales et modifiées, un chronomètre, un ou plusieurs compteurs de différences trouvées, et une section de messages où différents événements de partie sont affichés. Tous ces éléments doivent être complètement visibles dans la vue, et ce en tout temps. En aucun cas le joueur ne devrait avoir à faire défiler la page ou redimensionner la fenêtre pour révéler son contenu.

*Note : Considérez que l'utilisateur aura un écran d'une résolution d'au moins 1920 x 1080 et que le site web sera vu en mode plein écran.*

#### Chronomètre

Le chronomètre est là pour indiquer au joueur le temps passé depuis le début du jeu. Il est initialisé à zéro et est arrêté dès que la partie est terminée. Il doit être affiché selon le format suivant : *MM:SS* (minutes et secondes).

#### Compteur de différences trouvées

Lors d'une partie en cours, pour chaque joueur il doit y avoir un compteur de différences trouvées. Comme son nom l'indique, ce compteur affiche le nombre de différences trouvées par le joueur et sa valeur est incrémentée à chaque fois qu'il en trouve une nouvelle. Puisque le nombre de différences trouvées ne sera jamais de dix ou plus, le compteur est affiché sur un seul caractère numérique.

### Section de messages

Durant une partie, des messages sont envoyés aux joueurs à la suite de différents événements. Certains d'entre eux ont lieu dans la partie en cours, d'autres sont générés par des joueurs ailleurs sur le site. La présentation des messages est comparable à celle d'une boîte de clavardage. C'est-à-dire que les nouveaux messages apparaissent dans le bas de la boîte de texte et les anciens sont poussés vers le haut. De plus, lorsque les messages plus anciens tombent « hors champ » une barre de défilement apparaît sur la boîte permettant ainsi au joueur pouvoir revoir tous les messages. La boîte de messages est de taille fixe. Elle ne peut pas être redimensionnée ni masquée.

Les messages en tant que tels sont décrits dans leur propre section de ce document, sous *Description des messages*.

### Identification d'une différence

Pour identifier une différence, le joueur n'a qu'à cliquer là où il en voit une. Il peut cliquer tant sur l'image originale que sur la modifiée. Si le pixel cliqué appartient à une différence n'ayant pas encore été identifiée, celle-ci est alors considérée comme trouvée et est restaurée à son état d'origine. C'est-à-dire que chaque pixel de la différence sur l'image modifiée prend la couleur du pixel correspondant sur l'image originale. Un effet sonore est joué et le compteur de différences trouvées du joueur ayant fait la découverte est augmenté de un.

Dans une partie en mode *points de vue multiples* les sept différences de la première image modifiée sont les mêmes que celles de la deuxième image modifiée. Elles sont simplement observées d'un autre angle. Bien que ce soit les « mêmes » différences, chacune d'elle doit être identifiée individuellement. Autrement dit, identifier une différence dans un point de vue ne fait pas révéler la différence correspondante dans l'autre point de vue.

### Erreur d'identification

Lorsqu'un joueur clique dans une image là où il n'y avait pas de différence, un effet sonore est joué. De plus, le mot « ERREUR » apparaît à l'endroit du clic et reste visible durant une seconde. Pendant ce délai, le pointeur de souris du joueur doit être remplacé par un symbole d'interdiction et ses clics doivent être ignorés. Il s'agit d'une forme de pénalité pour décourager le joueur à cliquer partout sans arrêt.

Dans une partie à deux joueurs, seul le joueur ayant commis l'erreur voit le mot et entend l'effet sonore.

Pour être plus visible, le mot « ERREUR » doit être d'une couleur claire et entouré d'une bordure noire.

## Vue d'administration<sup>2</sup>

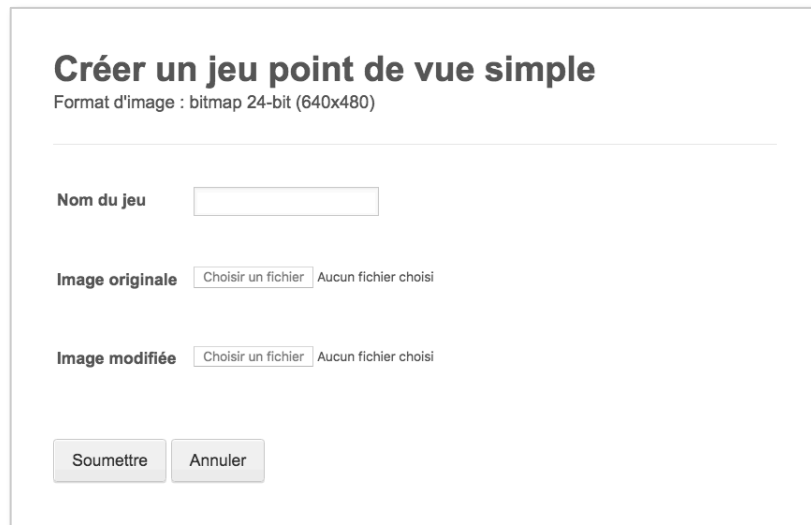
C'est ici que l'administrateur du site peut gérer son contenu. Les actions qu'il peut entreprendre sur cette page sont les suivantes :

- Créer un jeu en mode *point de vue simple*
- Créer un jeu en mode *points de vue multiples*
- Supprimer un jeu
- Réinitialiser les meilleurs temps d'un jeu

La vue d'administration est presque identique à la vue de sélection. C'est-à-dire que les fiches de jeu y sont présentées, et ce de la même manière que sur la vue de sélection. Une différence est qu'en haut de la vue se trouve une bande fixe contenant deux boutons. Un pour la création de jeu en mode *point de vue simple* et l'autre pour la création de jeu en mode *points de vue multiples*. Étant fixe, la bande doit rester en place lorsque l'utilisateur fait défiler la page.

### Créer un jeu en mode *point de vue simple*

Lorsque l'administrateur clique sur le bouton de création pour ce mode, un formulaire superposé à la page apparaît. Il permet d'y entrer le nécessaire pour la création d'un nouveau jeu. Soit un nom et deux images : l'originale et la modifiée. Une fois le formulaire rempli, l'administrateur clique sur *Soumettre* pour envoyer sa requête ou *Annuler* pour faire disparaître le formulaire. Ci-bas un exemple de formulaire avec les champs requis.



Créer un jeu point de vue simple

Format d'image : bitmap 24-bit (640x480)

Nom du jeu

Image originale  Aucun fichier choisi

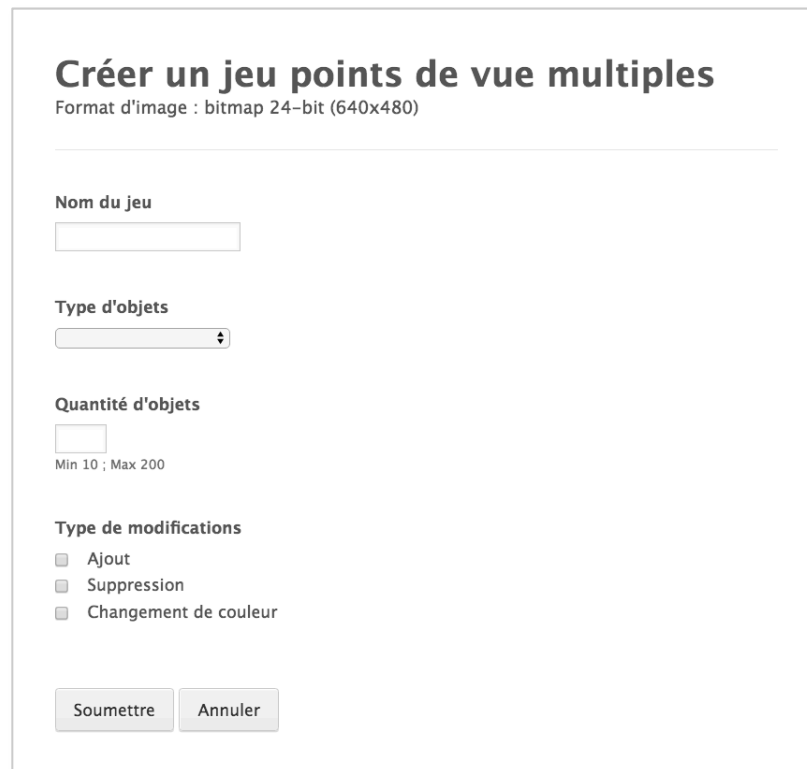
Image modifiée  Aucun fichier choisi

Exemple d'un formulaire de création (point de vue simple)

<sup>2</sup> Doit être accessible à <http://<adresse du serveur>/admin>. L'accès est libre et **n'est pas** protégé par un mot de passe.

### Créer un jeu en mode *points de vue multiples*

L'idée est la même pour le mode *points de vue multiples*. L'administrateur doit remplir les champs d'un formulaire et cliquer sur *Soumettre* pour envoyer sa demande de création. Le menu déroulant de type d'objets doit avoir les options suivantes : « formes géométriques » et « thématique ». Pour ce qui est du type de modifications, au moins une option doit être cochée. Il est aussi possible de choisir une combinaison de deux options et même les trois options.



The screenshot shows a web form titled "Créer un jeu points de vue multiples" with a subtitle "Format d'image : bitmap 24-bit (640x480)". The form contains the following fields and options:

- Nom du jeu**: A text input field.
- Type d'objets**: A dropdown menu with a downward arrow.
- Quantité d'objets**: A text input field with a range "Min 10 ; Max 200" below it.
- Type de modifications**: A section with three checkboxes:
  - ☐ Ajout
  - ☐ Suppression
  - ☐ Changement de couleur
- At the bottom, there are two buttons: "Soumettre" and "Annuler".

*Exemple d'un formulaire de création (points de vue multiples)*

*Note : le fonctionnement de la création de jeux est détaillé plus loin dans des sous-sections de « Composition d'un jeu ».*

En cas d'erreur lors du remplissage d'un formulaire, l'utilisateur doit en être informé puis se faire guider dans la résolution du problème.

### Supprimer et réinitialiser un jeu

L'autre différence entre les vues de sélection et d'administration concerne les fiches. Sur la vue d'administration, le bouton *jouer* de chaque fiche doit être remplacé par un bouton *supprimer* et le bouton *joindre* ou *créer* doit être remplacé par un bouton *réinitialiser*.



Cliquer sur *supprimer* fait effacer la fiche de jeu tandis que cliquer sur *réinitialiser* remplace les meilleurs temps (et leurs auteurs) par des valeurs fictives. Vu le caractère permanent de ces actions, un message de confirmation est présenté à l'administrateur avant d'exécuter la requête.

*Attention : La suppression d'une fiche de jeu doit être gérée de sorte que les utilisateurs en cours d'interaction avec le jeu n'aient pas de problèmes. Les parties en cours doivent donc se poursuivre et se terminer normalement et un joueur en attente d'adversaire est renvoyé à la vue de sélection avec un message lui expliquant que le jeu n'est plus disponible.*

## Description des messages

Les messages doivent respecter un format précis où

- *HH:MM:SS*<sup>3</sup> indique à quelle heure l'évènement a eu lieu. Il faut afficher les heures, minutes et secondes. Le heures sont sur un format 24h.
- *NOM\_JOUEUR* indique quel joueur à déclencher l'évènement.
- *NOM\_JEU* est le nom du jeu où l'évènement a eu lieu.
- *POSITION* est la position obtenue dans le tableau des meilleurs temps. Peut être : « première », « deuxième » ou « troisième ».
- *NB\_JOUEURS* devient « solo » ou « un contre un » dépendamment du type de partie jouée.

Les deux évènements suivants génèrent des messages qui ne sont affichés qu'au(x) joueur(x) de la partie où ils sont déclenchés.

### Différence trouvée

Lorsqu'une différence est trouvée, un message est affiché. S'il s'agit d'une partie *solo*, le message sera :

*HH:MM:SS* – Différence trouvée.

S'il s'agit d'une partie *un contre un* le message sera plutôt :

*HH:MM:SS* – Différence trouvée par *NOM\_JOUEUR*.

### Erreur d'identification

Lorsqu'un joueur clique pour identifier une différence, mais que c'est une erreur et qu'il n'y en a pas. S'il s'agit d'une partie *solo*, le message sera :

*HH:MM:SS* – Erreur.

---

<sup>3</sup> Il s'agit ici de l'heure de la journée. **À ne pas confondre avec la valeur du chronomètre.**

S'il s'agit d'une partie *un contre un* le message sera plutôt :

*HH:MM:SS* - Erreur par *NOM\_JOUEUR*.

Les trois événements suivants génèrent des messages qui sont affichés à tous les joueurs se trouvant sur le site.

#### *Meilleur temps*

Lorsqu'un joueur termine ou gagne une partie avec un temps le plaçant parmi les trois meilleurs sur une fiche de jeu.

*HH:MM:SS* - *NOM\_JOUEUR* obtient la *POSITION* place dans les meilleurs temps du jeu *NOM\_JEU* en *NB\_JOUEURS*.

#### *Connexion*

Lorsqu'un nouveau joueur est arrivé sur le site. Plus particulièrement, cet événement est déclenché lorsqu'un utilisateur a donné un nom de joueur et que celui-ci a été validé.

*HH:MM:SS* - *NOM\_JOUEUR* vient de se connecter.

#### *Déconnexion*

Lorsqu'un joueur a quitté le site.

*HH:MM:SS* - *NOM\_JOUEUR* vient de se déconnecter.

### Gestion et déroulement des parties

Le système doit être capable de supporter l'exécution de plusieurs parties en même temps, et ce pour chaque jeu disponible. Il doit aussi se charger du jumelage des joueurs des parties en *un contre un*.

#### *Jumelage*

Les joueurs n'ont pas la possibilité de choisir leur adversaire lorsqu'ils demandent de jouer en *un contre un*. Un système automatisé a la tâche de les jumeler sur la base du « premier arrivé, premier servi ».

#### *Logique de jumelage*

Lorsqu'il n'y a aucun joueur en attente d'adversaire, le bouton d'*un contre un* affiche *créer*. Un joueur cliquant sur *créer* tombe aussitôt en attente d'adversaire. Le bouton affiche alors *joindre*. Dès qu'un autre joueur clique sur *joindre*, il y a jumelage entre les deux joueurs et une partie commence entre eux. Le bouton redevient *créer*.

Si pendant son attente un joueur annule sa demande de création de jeu, il est renvoyé à la vue de sélection de partie et le bouton redevient *créer*.

Un jumelage ne peut être fait que pour deux joueurs demandant de jouer une partie en *un contre un* **sur une même fiche de jeu**.

*Note : Le changement de l'inscription du bouton, « créer » ou « joindre », doit se faire en temps réel de sorte que tous les joueurs déjà présents dans la vue de sélection voient le changement s'effectuer devant eux. Ils ne doivent surtout pas avoir à rafraichir la page pour voir le changement.*

### Initialisation

Avant que la vue de jeu soit présentée au(x) joueur(s), les différents éléments de jeu doivent être initialisés à leur état de départ. Plus précisément, le compteur de différences de chaque joueur et le chronomètre doivent être mis à zéro. De plus, la section de messages d'évènements doit être vide.

### Jeu

Dès que la vue est présentée au(x) joueur(x), la partie commence. Le chronomètre est alors démarré et les joueurs peuvent cliquer sur les images.

En *un contre un* les joueurs jouent en même temps sur le même jeu. La restauration à son état d'origine d'une différence trouvée est donc visible pour les deux joueurs.

### Fin de partie

Une partie se termine lorsqu'un nombre précis de différences trouvées est atteint. Dans une partie *solo*, le joueur doit en trouver 7 en mode *point de vue simple* ou 14 en mode *point de vue multiples* (donc toutes les différences). En *un contre un*, la partie se termine **dès qu'un des deux joueurs atteint** 4 différences trouvées en *point de vue simple* ou 7 en *points de vue multiples*.

Une fois le seuil atteint, le chronomètre est arrêté, tout clic sur une image est ignoré et un message de félicitation est affiché. En *un contre un* le message de félicitation n'est vu que par le gagnant. Le perdant voit plutôt un message l'invitant à faire une nouvelle partie. Lorsque le joueur ferme le message, il est ramené à la vue de sélection de partie.

### Mise à jour des meilleurs temps

La fin de partie provoque aussi la mise à jour du tableau des meilleurs temps. Si un nouveau meilleur temps est obtenu, celui-ci et son auteur sont inscrits au tableau approprié. Quand deux joueurs sont ex æquo, celui qui était déjà présent au tableau conserve la meilleure place.

## Composition d'un jeu

Un jeu en *point de vue simple* contient un trio d'images et un jeu en *points de vue multiples* en contient deux, soit un par point de vue. Un trio d'images est composé de :

- Une image originale
- Une image modifiée
- Une image de différences

L'originale est une image quelconque qui sert de base au jeu.

La modifiée est similaire à l'originale à l'exception de sept différences obtenues après l'application de changements.

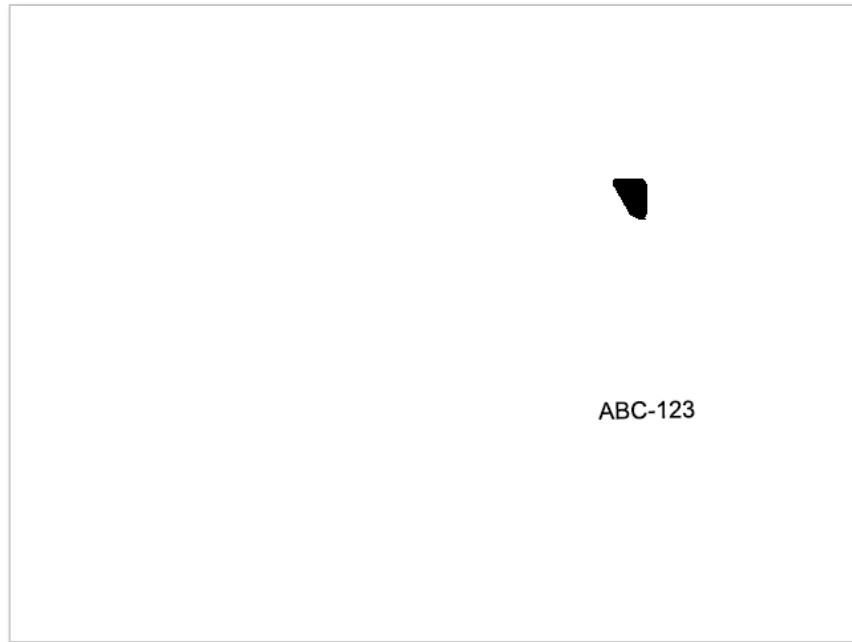
Celle de différences est une image en noir et blanc qui sert à montrer où se trouvent les différences entre les deux images. Elle est utilisée par le jeu pour rapidement déterminer si le joueur a trouvé une différence lors d'un clic. Elle est aussi utile pour faire un contrôle visuel lors de la création des jeux, ce qui aide au débogage. Le système la construit en deux étapes à l'aide des images originale et modifiée.

Pour faire un exemple, prenons les images suivantes où nous avons apporté deux changements. Soit le retrait d'un miroir (côté conducteur) et l'ajout de « ABC-123 » sur la plaque d'immatriculation.



*Images originale (gauche) et modifiée (droite)*

Ensuite, créons l'image de différences. La première étape consiste à identifier tous les pixels modifiés. Pour chaque pixel n'ayant pas exactement la même valeur de couleur sur les deux images, un pixel noir est dessiné à la même position sur l'image de différence. Pour ceux ayant la même valeur, le pixel correspondant est plutôt dessiné en blanc.



*Image de différences – première étape*

Avant de passer à la deuxième étape, voyons ce qu'est une « différence ».

#### Définition d'une différence

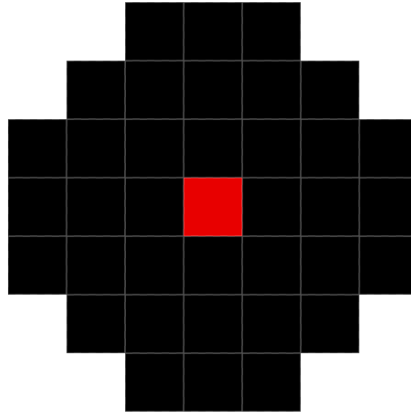
Pour qu'une différence soit considérée comme telle, celle-ci doit être en un seul « morceau ». Autrement dit, tous les pixels modifiés par le changement doivent former un groupe où pour chaque pixel il existe un chemin de pixels menant à tout autre pixel du groupe. Par exemple l'ajout d'un caractère « R » sur une image est considéré comme une différence, car tous les pixels sont liés ensemble. L'ajout d'un « i » n'est pas considéré comme une différence, mais deux. En effet, la barre et le point ne se touchent pas et sont donc considérés comme deux différences distinctes. La même idée s'applique pour les chaînes de caractères. Par exemple l'ajout d'une chaîne contenant « xyz » produit trois différences.

Pour que deux pixels soient considérés comme se touchant, il faut qu'ils soient directement l'un à côté de l'autre ou en diagonale. Par exemple, dans une matrice de 3 x 3 pixels, celui au centre est considéré comme touchant les huit autres qui l'entourent.

Revenons à notre exemple où nous souhaitons créer deux différences : le miroir et l'immatriculation. Selon la définition d'une différence, nous avons un problème puisque l'ajout du texte est considéré comme non pas une, mais bien sept différences. Soit, une par caractères.

Pour réduire la fréquence d'occurrence ce type de problème et par la même occasion rendre les petites zones de différences plus facilement cliquables par le joueur, il suffit

d'élargir la zone considérée comme modifiée à l'entour de chaque pixel différent. C'est ce qui constitue la deuxième étape dans la création de l'image de différences. Pour chaque pixel en noir obtenu à la première étape, 36 pixels noirs sont ajoutés à l'entour de lui selon la disposition illustrée ci-dessous.



*Élargissement d'un pixel de différence*

*Note : le pixel d'origine est en rouge seulement pour le différencier de ceux qui lui sont adjoints. Il est normalement en noir sur l'image de différence.*

Voici l'image de différences finale, après l'élargissement de chaque pixel noir.



*Image de différences – deuxième étape*

Nous voyons que désormais tous les pixels de la chaîne de texte se touchent. La chaîne forme donc une seule différence tel qu'initialement souhaité.

Même si c'est loin d'être l'idéal dans un contexte d'application web, **toutes les images composant les jeux doivent être dans un format *bitmap* (BMP) 24-bit et avoir une taille de 640 x 480 pixels**. Ce choix est motivé par la simplicité du format BMP.

#### Exécutable générateur d'images de différence

Pour faciliter les tests, la procédure de création d'une image de différences doit être sous forme d'un exécutable séparé. Ainsi, il sera autant possible de le lancer manuellement pour le tester que de le voir appelé programmatiquement par d'autres composants du système.

La syntaxe à utiliser est la suivante :

```
bmpdiff [-partiel] <original> <modifié> <sortie>
```

Les arguments *original*, *modifié* et *sortie* sont respectivement le nom du fichier de l'image original, le nom du fichier de l'image modifiée et le nom désiré pour le fichier de différences à créer.

Si utilisée, l'option *-partiel* fait en sorte que l'étape d'élargissement des pixels est ignorée avant la production de l'image de différences. Cette option ne doit être utilisée que manuellement dans un contexte de test ou de débogage.

En reprenant l'exemple du véhicule *Nissan Patrol*, utiliser

```
bmpdiff -partiel nissan.bmp nissan_modifié.bmp nissan_diff.bmp
```

produirait une image de sortie nommée *nissan\_diff.bmp* où l'on voit clairement l'inscription « ABC-123 ». (image de différence - première étape)

#### Création d'un jeu en point de vue simple

Tel qu'expliqué dans la section *Vue d'administration*, l'utilisateur crée un nouveau jeu *point de vue simple* en soumettant une paire d'images et un nom de jeu.

À la réception de ces images, le système doit produire l'image de différences et la valider. Si celle-ci ne comporte pas exactement sept différences, la création du nouveau jeu est annulée et l'utilisateur ayant fait la requête en est informé. Lorsqu'un jeu est validé, les éléments suivants doivent être générés :

- Tableau des meilleurs temps en mode *solo*
- Tableau des meilleurs temps en mode *un contre un*
- Vignette de l'image originale

Ces trois éléments ainsi que le nom donné sont utilisés pour construire une nouvelle fiche de jeu qui sera désormais disponible dans la *Vue de sélection*.

*Note : Puisqu'il n'y a encore eu aucune partie de jouée avec le nouveau jeu, il faut remplir les tableaux de meilleurs temps avec des entrées fictives. Les temps fictifs doivent être assez élevés de sorte qu'ils soient faciles à battre.*

### Création d'un jeu en points de vue multiples

Pour la création d'un jeu en *points de vue multiples* l'utilisateur doit fournir un nom, un type d'objets, une quantité d'objets et un ou plusieurs types de modifications.

Le nom ne sert qu'à identifier le jeu à créer tandis que les autres valeurs servent à paramétrer la création des images. En effet, contrairement au jeu en *point de vue simple* les images ne sont pas fournies par l'utilisateur, mais plutôt générées par un programme.

### Génération d'images

En *points de vue multiples* il y a deux images originales au lieu d'une seule, et il en est de même pour les images modifiées correspondantes.

Le programme générateur d'images a pour tâches de créer une scène 3D, la remplir d'objets, la photographier à partir deux points de vue différents<sup>4</sup>, modifier son contenu, et finalement la photographier à nouveau à partir des mêmes points de vue.

### Construire la scène 3D

Deux types de scènes peuvent être créées. Il y a celle contenant uniquement des formes géométriques solides et celle contenant des objets respectant un thème. Le choix du type est fait sur le formulaire de création.

Une scène contenant des formes géométriques n'a pas de décor particulier autre qu'une couleur de fond unie. Elle est remplie d'objets choisis au hasard parmi les suivants :

- Sphère
- Cube
- Cône
- Cylindre
- Pyramide à base triangulaire

---

<sup>4</sup> La caméra est située à une position différente pour chaque photo, mais son point visé reste le même.



La position, l'orientation et la couleur des objets doivent être variées et aléatoires. La taille des objets doit aussi être choisie au hasard, mais doit se situer entre 50% et 150% d'une taille de référence. La scène doit être éclairée de sorte qu'il soit possible de clairement distinguer les objets dessinés et leurs différentes faces.

La scène thématique doit quant à elle avoir un décor en lien avec le thème des objets. Par exemple, si le thème est *la nature*, le décor pourrait être constitué d'un sol couvert de diverses herbes, d'un ciel ensoleillé et de la présence d'une chaîne de montagnes au loin. Le thème choisi doit offrir au moins 15 sortes d'objets différents. Dans notre exemple, ça pourrait être des plantes ou des animaux. Même si le choix des objets ajoutés est aléatoire, cela doit rester cohérent avec le thème choisi. Le pourcentage de représentation des différents types d'objets doit être logique. Par exemple, une « forêt » avec un seul arbre et 199 chevreuils n'a pas vraiment de sens.

*Note : Le thème n'est pas imposé. Il revient aux développeurs de le choisir.*

Le placement des objets sur la scène est aléatoire, mais doit respecter la logique du thème. Par exemple, si un des objets est un arbre, celui-ci devrait être ancré au sol et non flotté dans les airs! Il en va de même pour l'éclairage qui doit aussi tenir compte du thème et de la composition de la scène.

Le nombre d'objets à placer dans la scène doit être compris entre 10 et 200. Cette valeur est spécifiée par l'utilisateur sur le formulaire de création.

#### *Générer les images originales*

Pour une scène d'objets géométriques, le point de vue de chacune des photos n'a pas tellement d'importance puisque le sujet est abstrait. Dans le cas d'une création thématique, les points de vue doivent permettre d'apprécier la scène. Pour reprendre l'exemple de la nature, un point de vue de face ou légèrement en angle vers le bas depuis la cime d'un arbre serait intéressant.

#### *Modifier la scène et générer deux nouvelles images*

Il existe trois types de modifications pour créer les sept différences par image. Il y a l'ajout, la suppression et le changement de couleur d'un objet. La routine de modification doit donc s'exécuter à sept reprises. À chaque fois, le type de modification à appliquer est choisi au hasard par ceux spécifiés par l'utilisateur sur le formulaire de création.

Pour un objet « thématique », le changement de couleur peut être remplacé par un changement de texture.

*Attention : une modification peut parfois être visible d'un point de vue, mais pas d'un autre. De même qu'une modification peut entraîner plus d'une différence. Par exemple,*

*l'ajout d'un objet derrière un autre qui le sépare en deux. Il faut donc développer la génération d'image en tentant de réduire le plus possible le risque de manifestation de ce type de problème. Finalement, il faut faire attention à ce qu'une modification ne soit pas appliquée à un objet issu d'une autre modification. Par exemple, supprimer un objet qui vient d'être ajouté!*

Un coup les modifications appliquées à la scène, celle-ci est photographiée à nouveau à partir des mêmes points de vue que précédemment.

#### *Valider les images modifiées*

Les images originales et modifiées étant maintenant générées, le système produit les deux images de différences. Il s'assure ensuite que chacune d'elle contient bel et bien sept différences. Si ce n'est pas le cas, une nouvelle génération d'images est lancée avec les mêmes paramètres. Jusqu'à quatre essais doivent être faits avant de laisser tomber. Après quatre échecs, la création du nouveau jeu est annulée et l'utilisateur en est notifié. Il peut alors changer ou non les paramètres de création et essayer à nouveau.

Si le jeu est toutefois valide, la création est finalisée de la même manière que pour un jeu en *point de vue simple*. C'est-à-dire que les tableaux de meilleurs temps sont générés et qu'une vignette de la première image originale est produite. La fiche de jeu est ensuite rendue disponible dans la *Vue de sélection*.

#### *Exécutable générateur d'images*

Le programme générateur d'images doit être contenu dans un exécutable séparé. Comme pour l'autre exécutable décrit précédemment, celui-ci pourra être appelé manuellement ou programmiquement.

La syntaxe à utiliser est la suivante :

```
genmulti {geo|theme} <quantité> <modification>... <sortie>
```

Il faut obligatoirement spécifier *geo* ou *theme* pour indiquer si la scène comporte des formes géométriques ou des objets thématiques. Pour *quantité*, une valeur entre 10 et 200 doit être fournie. *modification* spécifie quelles modifications peuvent être appliquées. Il faut entrer « a » pour l'ajout, « s » pour la suppression et « c » pour le changement de couleur. Il faut spécifier un, deux ou tous les types de modifications. Finalement, *sortie* servira de préfixe aux noms des fichiers de sorties qui auront la forme suivante : *sortie\_a\_ori.bmp*, *sortie\_b\_ori.bmp*, *sortie\_a\_mod.bmp*, *sortie\_b\_mod.bmp*. Les fichiers « a » pour le premier point de vue et les « b » pour le second. « ori » pour image originale et « mod » pour image « modifiée ».

Par exemple,

```
genmulti geo 180 ac test
```

Créera une scène composée de 180 formes géométriques. Chaque modification appliquée sera choisie entre un ajout d'objet ou le changement de couleur d'un objet.

L'image originale du premier point de vue se nommera : *test\_a\_ori.bmp*.

L'image originale du deuxième point de vue se nommera : *test\_b\_ori.bmp*.

L'image modifiée du premier point de vue se nommera : *test\_a\_mod.bmp*.

L'image modifiée du deuxième point de vue se nommera : *test\_b\_mod.bmp*.