

JAVA SE

PROJET « Jeux d'enfants »

Table des matières

1	Présentation	2
2	Contraintes techniques	2
3	Livrables attendus	3
4	Pour aller plus loin.....	3
5	Memento - Aide technique	3
6	Contraintes de la présentation.....	4

1 Présentation

Vous devez réaliser un programme « multi-jeux » à destination des jeunes enfants. Ce programme propose trois activités :

- Une ardoise « magique » pour dessiner à volonté,
- Des exercices de calcul,
- Un jeu de questions-réponses.

2 Contraintes techniques

Le programme est entièrement graphique. Il se tiendra dans une fenêtre unique qui contiendra quatre activités :

- La première activité, nommée *Dessin*, affichera l'ardoise magique avec une barre d'outils proposant un bouton pour tout effacer et une sélection de couleur (rouge, vert et bleu *a minima*).
- La seconde activité, nommée *Calcul*, proposera un calcul aléatoire à résoudre, un champ de saisie pour la réponse (champ ayant automatiquement le focus lors de l'affichage du calcul proposé) et des boutons « *Vérifier* », « *Solution* » et « *Autre calcul* ».
- Le troisième onglet, nommé *Question*, affichera une question tirée aléatoirement parmi celles présentes dans la base de données, un champ de réponse (champ ayant automatiquement le focus lors de l'affichage de la question) et les boutons *Vérifier*, *Solution* et *Autre question*.
Attention : il est nécessaire que la réponse soit juste quelle que soit la casse utilisée, et surtout que les accents soient utilisés ou non. Nous ignorerons la réforme de l'orthographe pour ce sujet...
- Le quatrième onglet, nommé « *Administration* », sera invisible par défaut. Il ne sera visible que si l'on donne le mot de passe administrateur du logiciel. Cet onglet doit permettre de modifier ou d'ajouter une question à la base de données. Il affichera donc :
 - Une sélection du niveau de la question (1 ou 2) par bouton radio,
 - Le numéro de la question par menu déroulant dans le cas d'une demande de modification,
 - Le champ d'édition de l'énoncé de la question,
 - Le champ d'édition de la réponse à la question,
 - Un bouton « *Enregistrer* » dans le cas d'une création, « *Modifier* » dans le cas d'une modification.
- Le mot de passe de l'administrateur ne doit pas pouvoir être lu directement dans le code source, ni même au niveau de la machine.

La fenêtre principale proposera également un menu contenant :

- Une entrée *Activité* proposant les trois activités. Le clic sur l'une d'elle sélectionnera l'onglet correspondant.
- Une entrée *Niveau* proposant deux niveaux de difficulté (en option selon le temps).
- Une entrée *Administration* proposant la modification de la base de données des questions (ajouter, modifier ou supprimer une question). L'accès à ce menu sera soumis à l'entrée d'un mot de passe. L'accès, une fois autorisé, affiche le quatrième onglet dans la fenêtre.

Les exercices de calculs seront créés à la demande et dynamiquement par le programme lui-même. Pour les calculs comme pour les questions, les contraintes sont les suivantes :

- Si le niveau est réglé sur 1, alors le programme propose :

- Pour l'onglet *Calcul*, des additions et des soustractions de nombres positifs à 1 chiffre, le résultat devant être positif (mais pouvant être à 2 chiffres).
- Pour l'onglet *Question*, des questions de niveau 1 (très simples).
- Si le niveau est réglé sur 2, alors le programme propose :
 - Pour l'onglet *Calcul*, des additions et des soustractions de nombres positifs jusqu'à 3 chiffres, le résultat pouvant être négatif ou sur 4 chiffres, ou des multiplications de nombres positifs à 1 chiffre.
 - Pour l'onglet *Question*, des questions de niveau 1 ou 2.

Les questions et leurs réponses seront stockées dans une base de données MySQL éditée via une interface graphique (le quatrième onglet). Chaque question possède ainsi un niveau, un numéro, un énoncé et une réponse. La base de données se réduit donc à une unique table MySQL.

3 Livrables attendus

Les codes sources bien sûr, correctement rangés dans des packages adéquats, et correctement présentés (indentation correcte et commentaires au minimum, javadoc bienvenue).

Le plan de développement. Je ne vous demande pas de me rendre un devoir de gestion de projet, mais au moins le résultat de vos réflexions initiales présentant les choix faits, l'explication de ce choix, et les solutions techniques envisagées. Si les solutions techniques changent au cours du projet (ça arrive fréquemment), il faut expliquer pourquoi et les avantages de la nouvelle solution par rapport à l'ancienne... et prouver que ça n'affecte pas le reste du projet.

Bref, je veux voir les documents inhérents à la gestion de projet. Si en plus vous savez faire un Gantt et un prévisionnel de temps, ça n'en est que mieux, mais encore une fois ce n'est pas le but.

4 Pour aller plus loin

Si le projet vous semble trop court, vous pouvez améliorer les choses. Voici quelques idées :

- Programmer des touches de raccourci pour les différents menus et items de menu.
- Plus de forme de pointeurs pour le dessin (attention, ça peut grandement changer la façon de faire et remettre en question tout un pan de votre programme selon la solution choisie à l'origine).
- Pour le dessin toujours, pouvoir choisir plus de couleurs, voire toutes les couleurs.
- Rajouter de nouveau jeux (au hasard, un pendu par exemple, dont les mots sont stockés dans la base de données... ou un fichier) sur de nouveaux onglets. Bien sûr, il faudra ajuster les menus en conséquence. Et pourquoi pas un cumul de points avec déblocage de nouveaux jeux ?
- La possibilité de changer le mot de passe de l'administrateur.
- La possibilité de sauvegarder, charger et/ou redessiner un dessin.

Ce ne sont que quelques idées, les vôtres seront les bienvenues. Aucune obligation de réaliser ces idées, ce n'est que si vous désirez aller plus loin ET si vous en avez le temps !

5 Memento - Aide technique

- Un projet nécessite un *chef de projet*.
- Un projet est une succession de problèmes à résoudre, de TP à réaliser. La première chose à faire est donc de réaliser un plan de développement indiquant des solutions techniques. Chaque membre de l'équipe sera chargé de produire une partie du projet tirée de ce plan de

développement. Il est donc nécessaire que chacun apporte sa pierre à l'édifice, tant au niveau des idées de développement que des solutions techniques.

- Prenez un temps pour transmettre votre savoir et vos découvertes aux autres membres du projet.
- Les onglets dans une fenêtre s'obtiennent en utilisant un objet de type [JTabbedPane](https://docs.oracle.com/javase/7/docs/api/javax/swing/JTabbedPane.html) (<https://docs.oracle.com/javase/7/docs/api/javax/swing/JTabbedPane.html>) avec Swing ou [TabPane](https://docs.oracle.com/javase/8/javafx/api/javafx/scene/control/TabPane.html) (<https://docs.oracle.com/javase/8/javafx/api/javafx/scene/control/TabPane.html>) avec FX

D'autre part un dépôt Git est à votre disposition et contient un squelette du projet. Vous pouvez donc directement le cloner dans votre IDE et vous DEVREZ l'utiliser. Pour éviter les erreurs fréquentes dues à l'utilisation de Git, suivez ce plan :

- **Clonez** le dépôt sur votre machine via votre IDE,
- **Créez votre branche** de développement,
- Développez sur votre branche ,
- **Commit** multiples sur votre branche,
- Quand le développement de la fonctionnalité est terminé **checkout master**, **merge** votre branche, **pull** puis **push** sur le dépôt de référence (Gitlab),
- Avertir les autres qu'il y a eu **push** pour que les collègues fassent un **pull**
- **Créez une nouvelle branche** pour le prochain développement

Si vous suivez ces conseils, il n'y aura aucun problème.

L'autre possibilité est de faire « pro » : seul le chef de projet a le droit de **push**, les autres vont **cloner** le dépôt sur leur propre dépôt github/gitlab/bitbucket/autre, faire leurs modifications, puis faire un **pull request** pour que le chef de projet intègre leurs modifications. C'est plus lourd à gérer, mais c'est le chef de projet qui gère les intégrations des nouveautés.

6 Contraintes de la présentation

- Prenez le temps de préparer votre présentation. Un bon *Power Point* est toujours appréciable.
- Le chef de projet fera la présentation globale du projet.
- Chaque membre du groupe devra présenter sa partie. Donc tout le monde va parler...
- Chaque membre du groupe devra faire une présentation partiellement en anglais (plus de 3 mots, merci !), ceci afin de vous aguerrir pour votre vie professionnelle future.
- La présentation durera au plus une heure tout compris. Je compte :
 - 15 minutes maximum d'installation et de préparation.
 - 20 à 30 minutes de présentation orale.
 - 15 à 20 minutes d'échanges avec moi d'abord dans le rôle du patron, puis un jeu de questions-réponses avec les autres spectateurs (les autres stagiaires) ensuite, ceci afin d'éclaircir des points techniques ou des choix effectués.
- Comme il est hors de question de vous laisser aller voir le client directement, la présentation sera faite comme si votre interlocuteur était votre patron. Vous devrez donc défendre votre projet tant au niveau de l'ergonomie que de la programmation (choix de développement, méthode de travail utilisée, concepts mis en œuvre, exemples de code...). Ne vous enfoncez pas dans le code avec une analyse ligne à ligne, ça ne présente aucun intérêt à part indiquer que vous ne savez pas quoi dire sur votre propre travail...