

Projet pédagogique AP2 - Diplodocalcul

Introduction

Contexte

Ce logiciel est créé dans le cadre d'une collaboration avec l'école primaire et maternelle Pierre et Marie Curie situé à Villeneuve d'Ascq. Il est réalisé dans le cadre du contrôle continu du module d'algorithmique et programmation du DUT Informatique de l'IUT « A » de l'université Lille 1.

L'ensemble de la promotion a été divisé en binôme et devait réaliser un cahier des charges et un logiciel ludo-éducatif.

But du projet

Ce projet vise à produire un logiciel éducatif destiné à des enfants de 6 à 11 ans afin de les accompagner dans l'apprentissage des mathématiques.

Grâce à une difficulté contrôlée et des découpages par tranche d'âge, il est très accessible.

Les tranches en question sont les suivantes :

- Cours préparatoire : entre 6 et 7 ans
- Cours élémentaire 1 et 2 : entre 7 et 9 ans
- Cours moyens 1 et 2 : entre 9 et 11 ans

Description de la demande

Objectifs

L'objectif de ce logiciel est d'accompagner l'enfant dans son apprentissage des mathématiques et de vérifier sa compréhension de celles-ci.

Afin de capter l'attention des plus jeunes, l'utilisation d'illustrations et d'animations est prévue.

Présentation du logiciel

Le logiciel se présente, avant d'accéder aux calculs, en une succession de menu permettant de configurer la manière dont les calculs seront proposés. Tout d'abord, on demandera son prénom à l'utilisateur. Ensuite, on lui demandera, via une liste, de choisir son niveau d'étude. La dernière option à choisir concerne le mode de fonctionnement : apprentissage ou examen.

Une fois ces formalités accomplies, l'écran affichera selon le niveau, des images d'animaux/fruits/légumes ou directement des chiffres séparés par un opérateur.

L'élève devra sélectionner sa réponse parmi une liste de réponse.

Afin de s'adapter au plus grand nombre, la difficulté est croissante et reste constante tant que l'élève n'arrive pas à résoudre les calculs d'un niveau donné.

Critères de validation

Fonctions minimales

<https://diplodocalcul.googlecode.com>
Gavard Ewann (gavard.e@gmail.com)
Rabany Bertrand (rabany.b@gmail.com)

Le logiciel doit afficher une opération de façon aléatoire à résoudre. L'utilisateur doit pouvoir proposer une solution qui sera ensuite vérifiée par le logiciel puis corrigée si nécessaire.

Fonctions souhaitables

Il serait souhaitable, afin de permettre l'accès au plus grand nombre, d'adapter le niveau de difficulté des opérations en fonction du niveau sélectionné par l'utilisateur. De plus, afin de permettre à l'élève de se situer par rapport au reste de sa classe, un pourcentage de bonnes réponses ainsi qu'un score lui seront notifiés: une bonne réponse lui donne un point. Dans l'objectif de personnaliser l'affichage, on pourra demander le prénom et le nom de l'utilisateur.

Fonctions avancées

Tout d'abord un mode d'apprentissage sera proposé. Dans ce mode le nombre de calcul est illimité et le score affiché mais non enregistré. Le second mode lui, procède à l'enregistrement des scores et pourcentage de chaque élève qui sera accessible au professeur via une fonction dédiée.

Dans l'optique d'apporter de la convivialité au logiciel, des images pourront illustrer les calculs les plus faciles et des animations seront utilisées pour récompenser l'élève ou l'informer de son erreur.

Afin de permettre une interface plus intuitive, l'utilisation d'une liste de réponse cliquable, avec des réponses aléatoires et la bonne réponse peut être utilisée. Un clic suffirait à l'élève pour donner sa réponse et correspondrait d'autant plus aux élèves jeunes.

Analyse du logiciel

Contraintes techniques

Ce projet étant réalisé dans le cadre du module d'AP12, c'est l'*ijava*, une version simplifiée de Java qui sera utilisée par le biais du fichier *ap.jar*.

Contraintes temporelles

Le projet commence le 26 octobre par la présentation aux étudiants des objectifs de cet exercice.

Le projet est découpée en deux parties :

- la première concerne le cahier des charges.

Celui-ci est à rédiger par les élèves et doit être rendu avant le 31 Décembre 2011.

- la deuxième partie concerne le codage du projet en lui-même.

Le codage peut commencer dès le début du projet et doit être achevé pour le 19 Janvier 2012.

Planning du projet

Semaine 1 (9nov./ 16nov.) : Recherche du sujet du projet avec une visite sur différents sites web qui traitent du même sujet afin de trouver des idées supplémentaires et de pouvoir structurer le cahier des charges.

<https://diplodocalcul.googlecode.com>
Gavard Ewann (gavard.e@gmail.com)
Rabany Bertrand (rabany.b@gmail.com)

Semaine 2 (16nov./ 23nov.) : Approfondissement des recherches et début de rédaction du cahier des charges.

Semaine 3 (23nov./ 30nov.) : Poursuite de la rédaction du cahier des charges et début du codage du logiciel pour les fonctions de bases.

Semaine 4 (30nov./ 7 déc.) : Finalisation de la rédaction des parties du cahier des charges afin de pouvoir rendre la première partie et poursuite du codage du logiciel.

Semaine 5 (7déc./ 14 déc.) : Refonte du logiciel pour pouvoir prendre en compte les différentes exigences du logiciel.

Semaine 6 (14 déc./ 21 déc.) : Poursuite de la rédaction du cahier des charges et du codage du logiciel.

Semaine 7 (21 déc./ 28déc.) : Finalisation du cahier des charges.

Semaine 8 (28déc./ 2 jan.) : Amélioration du logiciel.

Semaine 9 (2 jan./ 9 jan.) : Poursuite du codage du logiciel afin de pouvoir atteindre les objectifs des « fonctions souhaitables » et rendu du cahier des charges.

Semaine 10 (9 jan./ 16 jan.) : Amélioration du code du logiciel et optimisation pour le rendu.

Analyse des données

Représentation de l'opérateur

Afin de permettre l'utilisation d'opérateurs adaptés à chaque tranche d'âge, il a été choisi de mettre les opérateurs dans un tableau de chaînes de caractères. Le niveau choisi permet, grâce à un *random()* modifié par un coefficient de sélectionner un opérateur valide.

Le tableau est constitué comme suit :

+ simple	- simple	* à un chiffre	+ compliqué	- compliqué	+ à virgule	- à virgule	/ entière	* compliqué	/ compliqué	* à virgule	/ à virgule
0	1	2	3	4	5	6	7	8	9	10	11

Génération d'opérandes

La génération d'opérandes est assurée par la fonction *random()* ; de *ijava*. La difficulté sert de coefficient.

Enregistrement des résultats

En mode « examen », les résultats de l'élève sont enregistrés. Ceux-ci, accessibles au professeur sont enregistrés dans un fichier .csv au nom de l'élève. Sa structure est la suivante :

<https://diplodocalcul.googlecode.com>
Gavard Ewann (gavard.e@gmail.com)
Rabany Bertrand (rabany.b@gmail.com)

ID	Prenom	Nom	Date	Niveau	Difficulté atteinte	Score
----	--------	-----	------	--------	------------------------	-------

Déroulement d'une session d'utilisation

Dans un premier temps, l'utilisateur est invité à saisir son Nom et son Prénom. Ensuite, son niveau lui est demandé. Enfin, l'utilisateur choisit son mode de jeu :

- 1) Illimité (Entraînement)
- 2) Examen

En mode illimité, l'utilisateur doit résoudre des problèmes de complexité croissante. L'exercice s'arrête à la demande de celui-ci. Dans le cas où l'utilisateur commet une erreur dans la résolution, la réponse lui est affichée et un calcul de difficulté identique lui est à nouveau proposé.

En mode examen, l'utilisateur doit répondre à dix calculs. La difficulté augmente cette fois beaucoup plus rapidement, et en cas d'erreur, elle continue à augmenter. Les résultats sont affichés en fin de séance. Ils sont également enregistrés afin d'être disponible pour le professeur.

Conclusion

Cette application a été réalisée malgré la difficulté d'obtenir par l'éducation nationale les acquis et compétences requises par les élèves pour chaque niveau. Nous avons donc choisi de synthétiser les informations disponibles sur internet.

Nous souhaitons que cette application puisse être utile au plus grand nombre et accompagner les élèves et leur professeur dans l'apprentissage des mathématiques.