

CS Games 2017



Programmation à relai I

Participants	3
Stations	3
Valeurs	7%
Durée	3 heures

Simulateur de développement logiciel

Vous venez d'être engagé dans une start-up de la Silicon Valley et vous voulez percer le domaine des jeux vidéo indépendant! Cette compagnie se spécialise dans la conception de simulateur et vous avez été assigné au nouveau projet nommé «Simulateur de développement logiciel».

Concept du jeu

Le jeu doit débuter la partie avec un budget de 100 000\$, à la semaine 0 et avec aucun employé. Le joueur doit pouvoir embaucher des ressources ou bien choisir une action. Par la suite, le joueur doit pouvoir avancer d'une semaine. Après chaque semaine, le jeu doit calculer le coût des ressources et le soustraire au budget. La progression du projet doit aussi augmenter selon les ressources alloué. Le but du jeu est de compléter le projet avant d'atteindre la faillite et ceci en moins de 52 semaines.

Condition de victoire


Le projet doit arriver à une progression de 100%, en moins de 52 semaines, avec un budget plus grand ou égale à 0.

Condition de défaite

- Dépassement du budget
- 52 semaines sont écoulées et la progression du projet n'a pas atteint 100%.

Prototype d'interface utilisateur

La conception de l'interface utilisateur a déjà été fait par un autre employé, vous devez vous baser sur ce prototype.



<div>Hire</div> <div>Programmer</div> <div>Tester</div> <div>Action(s)</div> <div>Outsource</div> <div>Early access release</div> <div>Next week</div> <div>Scoreboard</div> <div>Detailed Budget</div> <div>Quit</div>	<div>Week: 0 Progress : 0 % Budget balance: 100 000 Score: 0</div>
---	--

Objectifs

- ☐ Afficher la semaine courante (week #). (1 point)
- ☐ Afficher la balance du budget. (1 point)
- ☐ Afficher la progression courante du projet. (1 point)
- ☐ Afficher "x" programmeur(s) (image du bureau) dans l'espace de bureau. (3 points)
- ☐ Afficher "x" testeur(s) (image du bureau) dans l'espace de bureau. (3 points)

- ❑ Afficher le bouton "prochaine semaine" et implémenter l'avancement vers la prochaine semaine, imputation du budget et incrémentation de la progression. (5 points)
- ❑ Implémenter un écran annonçant la victoire ou la défaite. (3 points)
 - ❑ La partie recommence automatiquement après l'affichage du message, sans redémarrage du jeu. (2 points)
- ❑ Afficher le bouton et implémenter l'action pour ajouter un programmeur. (3 points)
 - Un programmeur coûte 1 200 par semaine.
 - Il fait progresser le projet de 1 point par semaine. Durant les 4 premières semaines suivant son embauche.
 - Ensuite, il fait progresser le projet de 1.4 point par semaine.
- ❑ Afficher le bouton et implémenter l'action pour ajouter un testeur. (3 points)
 - Un testeur coûte 800 par semaine.
 - Il fait progresser le projet de 0.4 point par semaine. Durant les 2 premières semaines.
 - Ensuite, il fait progresser le projet de 0.6 point par semaine.
- ❑ Afficher le bouton et implémenter l'action de sous-traiter du travail à bas prix. (3 points)
 - Coût de 2 000 et progression aléatoire, entre 1 et 6, non récurrent.
 - Peut-être utilisé 3 fois maximum durant une partie.
- ❑ Afficher le bouton et implémenter l'action pour faire une sortie de l'application en accès anticipé. (5 points)
 - Obtenez à l'instant, 15 000 de budget supplémentaire.
 - Par contre, la semaine suivante, la mesure coûte 1 000 et s'incrémente de 1 000 supplémentaire par semaine consécutive. Pour gérer la frustration des utilisateurs qui ont achetés en accès anticipé, en pensant qu'ils achetaient une solution complète.

- Donc, semaine #0 (accès anticipé) +15 000, semaine #1 -1 000, semaine #2 -2 000, etc.
- Peut-être utilisé seulement une fois.

☐ Implémenter un écran permettant de consulter le budget détaillé de la semaine, contenant les éléments suivants (5 points):

- Afficher la balance courante (montant du budget avant déduction)
- Coût des programmeurs
- Coût des testeurs
- Coût de la sous-traitance
- Coût ou budget supplémentaire provenant de l'accès anticipé
- Balance restante à la fin de la semaine

☐ Implémentation d'un système de pointage déterministe pour la partie. (2 points)

- ☐ Afficher le pointage courant dans le coin inférieur droit. (1 point)
- ☐ Offrir la possibilité d'entrer le nom du joueur à la fin de la partie et persister les pointages. (2 points)
- ☐ Ajouter un bouton pour afficher les pointages précédents persistés avec les noms des joueurs entrés précédemment. (3 points)

☐ Afficher et implémenter le bouton pour quitter le jeu. (2 points)

☐ Afficher le nom du projet en cours (1 point)

- ☐ Permettre au joueur de choisir le nom du projet (2 points)

☐ Afficher le nom de la compagnie dans l'espace de bureau (1 point) (au choix, 1 point bonus s'il y a un rapport avec votre thème d'équipe)

☐ Implémenter un système de triche pour avancer plus rapidement dans le jeu. (4 points)

- Documenter bien vos codes de triches dans un fichier "lisez-moi".

- ❑ Aligner parfaitement les images des programmeurs et testeurs, afin qu'il n'est pas d'espace entre les bureaux. (3 points)
- ❑ Inclure un fichier "lisez-moi" dans votre solution, documentant les objectifs complétés ainsi que le lancement de votre application. (1 point)
- ❑ Inclure un fichier "run.sh" pour démarrer votre solution. (1 point)