



Architecture MVC

OBJECTIF

Jusqu'à présent, pour répondre aux différentes questions des TD et TP, il vous était demandé de créer des fonctions dans des fichiers et de les "connecter" à des boutons. Un grand bond en arrière par rapport à tout ce qui a pu vous être enseigné en POO, notamment en ce qui concerne l'architecture d'une application.

Aujourd'hui, nous revenons à des principes de programmation plus élégants, mieux structurés et surtout beaucoup plus efficaces tant en terme de développement que de maintenance. Notamment avec la mise en place d'une architecture Modèle-Vue-Contrôleur ou MVC pour les intimes.

Rappel : Si des informations ne sont pas présentes dans ce sujet, c'est certainement qu'elles ont été données lors des TD précédents. Reportez-vous au sujet des TD 1, 2 et 3, au TP 1, ainsi qu'à vos notes. Pas de note ? Arf, c'est moche ce qui vous arrive...

Introduction

Vous dirigez une entreprise qui produit, à partir de matières premières et grâce aux savoir-faire de vos ouvriers, des vélos, scooters et voitures. Toute votre production est parquée dans un espace de stockage avant d'être vendue au consommateur. Ce dernier réalise ses achats en fonction de ses besoins qui sont très fortement influencés par les actualités du moment.

Pour gérer la production de votre usine, vous développez à vos heures perdues une application vous permettant de :

- commander vos matières premières
- recruter ou licencier du personnel
- produire des vélos, scooters et voitures
- observer l'état de vos stocks et de votre production, ainsi que les demandes des consommateurs

QUESTION 1

Les classes du modèle sont déjà développées, si ce n'est que la classe `Produit` n'a pas encore été spécialisée et qu'il manque quelques lignes de code par-ci, par-là.

- Créez trois fichiers `'velos.js'`, `'scooter.js'`, `'voiture.js'` dans lesquels vous déclarerez les classes `'Velo'`, `'Scooter'` et `'Voiture'` qui hériteront de la classe `Produit`.

	Stockage	Matières premières	Ressources humaines	Temps de fabrication	Prix de vente
Vélo	1	1	1	1 seconde	600 €
Scooter	3	5	2	2 secondes	3000 €
Voiture	10	20	10	8 secondes	15000 €

QUESTION 2

Le contrôleur est le chef d'orchestre de notre application. Il indique au modèle et à la vue ce dont il a besoin et ce qu'eux doivent faire. Le contrôleur est donc l'élément de l'application qui détermine ce qui se passe lorsque l'utilisateur clique sur un bouton ou complète un formulaire.

- Dans le constructeur de la classe 'Vue', ajouter un écouteur sur le document qui renverra tous les clics vers la fonction 'gererClick' du contrôleur (mais si, rappelez-vous, un truc du genre addEventListener...).

QUESTION 3

A chaque notification, le contrôleur demande à la vue d'actualiser les données affichées.

- Complétez la fonction 'actualiserRessources' qui indique en haut de l'écran les ressources dont dispose l'entreprise.
- Complétez la fonction 'actualiserProduction' qui indique à droite de l'écran la quantité de produits en cours de production.
- Complétez la fonction 'actualiserStock' qui indique la quantité de produits en stock (toujours à droite de l'écran).
- Enfin, complétez la fonction 'actualiserDemandes' qui affiche les demandes des consommateurs pour chaque type de produit.
- Testez.
- **Faites valider par l'enseignant les questions 1, 2 et 3 (5 points).**

QUESTION 4

Nous allons à présent mettre en place la commande des matières premières.

- Donnez le code de la fonction 'commanderMatierePremieres' de la classe 'Entreprise'. Une commande permet d'approvisionner l'entreprise avec 100 unités de matières premières. Chaque unité coûte 400 €.
- Créez une fonction 'commanderMatierePremieres' dans la classe 'Controleur'. Cette fonction donnera l'ordre à la classe 'Entreprise' de commander des matières premières.
- Complétez le constructeur de la classe 'Vue' afin d'associer le bouton 'btn-commander-matieres-premieres' à la fonction du contrôleur précédemment créée (la classe 'Controleur' possède une fonction 'ajouterEvenementClick').
- Testez (matières premières et trésorerie doivent être impactées).
- **Faites valider par l'enseignant (2 points).**

QUESTION 5

Vous avez compris le système ? La vue redirige les événements vers le contrôleur qui en fonction de l'action de l'utilisateur va dire au modèle et à la vue ce que chacun doit faire.

- Procédez de la même façon que précédemment afin de recruter et de licencier du personnel.

Attention : pour le licenciement, des indemnités d'un montant de 10 000€ seront versées à chaque employé licencié. Par ailleurs, un message d'erreur devra être affiché si l'on essaye de licencier alors qu'il n'y a plus de personnel dans l'entreprise.

Pour afficher un message d'erreur, utilisez le paramètre 'data' de la fonction 'notifier' :

```
this.notifier({ erreur : "Message d'erreur" });
```

QUESTION 6

La classe 'Entreprise' dispose d'une fonction 'produire' permettant de démarrer la production d'un produit.

- Donnez le code des fonctions 'produireVelo', 'produireScooter', 'produireVoiture' de la classe 'Entreprise'.
- Comme précédemment, liez ces fonctions à la vue et au contrôleur.
- Testez.
- **Faites valider par l'enseignant (2 points).**

QUESTION 7

La fonction 'produire' commence par tester si la production du produit est possible, puis, elle appelle la fonction 'demarrerProduction'.

- Consommer la matière première utilisée pour la fabrication du produit.
- Affecter le personnel nécessaire.
- Ajouter le produit à la production en cours.
- Puis, un fois le temps de fabrication du produit écoulé, elle appellera la fonction 'acheverProduction'.

La fonction 'acheverProduction' libère les ressources humaines employées à la fabrication du produit et place ce dernier dans le stock.

Attention : s'il n'y a plus de place dans le stock à ce moment, le produit est détruit et un message doit être affiché à l'utilisateur.

Pour temporiser l'appelle à une fonction :

```
var timer = setTimeout(function() { // code à executer }, tempsEnMillisecondes);
```

- Donnez le code des fonctions 'demarrerProduction' et 'acheverProduction'.
- Testez.
- **Faites valider par l'enseignant (4 points).**

QUESTION 8

Bien, à présent, entrons davantage dans le monde de la gestion d'entreprise. Premièrement, vous avez des salariés, il serait bon de les payer. Chaque employé vous coûte 2 500 € (charges comprises).

- Créez une fonction 'payerSalaires' dans la classe 'Entreprise' qui ampute votre trésorerie du montant nécessaire à la paie de vos salariés.
- Dans le contrôleur, mettez en place un ordre de paiement des salariés toutes les 15 secondes.
- Testez.
- **Faites valider par l'enseignant (2 points).**

Pour appeler une fonction périodiquement :

```
var timer = setInterval(function() { // code à executer }, tempsEnMillisecondes);
```

QUESTION 9

Vous fabriquez des produits, vous payez vos salariés, mais pour gagner de l'argent et rendre votre entreprise pérenne, il va falloir vendre ! Les consommateurs achètent en fonction de leurs besoins.

- Créez une fonction 'vendre' dans la classe 'Entreprise' qui prend en paramètre le type de produit à vendre, ainsi que la quantité. Cette fonction impactera les stocks et les ressources de l'entreprise.

Attention à bien prendre en compte les stocks disponibles.

- Dans le contrôleur, appelez la fonction 'acheter' des consommateurs toutes les 3 secondes. Cette fonction achète à l'entreprise donnée en paramètre, les produits en fonction des besoins des consommateurs.
- Testez.
- *Faites valider par l'enseignant (3 points).*

QUESTION 10

Les actualités, que les consommateurs lisent dans le journal ou entendent à la radio, influent sur les comportements.

La fonction 'getActualite' de la classe 'Actualités' permet d'obtenir une actualité au hasard.

- Modifiez le contrôleur afin d'obtenir régulièrement une actualité (toutes les 5 à 15 secondes) et de la faire connaître aux consommateurs (fonction 'apprendreActualite'). L'actualité obtenue devra être affichée comme alerte dans la vue.
- Testez.
- *Faites valider par l'enseignant (2 points).*