

Documentation UML

Diagramme UML Utilisation :

Les éléments du diagramme :

Acteur :

- Visitor
- User
- Factory

Use Case : Le diagramme de cas d'utilisation décrit les actions possibles des acteurs avec le système en question.

Cas d'utilisation décrits :

Log in :

- **Obligation des utilisateurs de se connecter pour valider la commande**

Sign up :

- **Processus d'inscription, l'utilisateur fournit ses données personnelles.**

Choose a vehicle :

- **Début du processus de commande. l'utilisateur choisi le modèle de voiture qu'il souhaite commander**

Customize it :

- **Deuxième étape du processus de commande.**
L'utilisateur choisit la couleur de la peinture de sa voiture, le type de moteur

Validate the order :

- **Valide la commande après avoir choisi le modèle de voiture et les paramètres désirés**

Track the progress of the order :

- **Met au courant l'utilisateur sur l'avancement de sa commande**

Receive notification of each change of step :

- **L'utilisateur reçoit une notification après chaque étape du processus fabrication de sa commande**

Receive notification in case of potential delays :

- **L'utilisateur reçoit une notifications s'il y a un potentiel retard sur sa commande**

put into production :

- **Après avoir effectué la commande, elle est envoyé à l'usine pour commencer la production du modèle**

Règles de gestion intégrés :

Le modèle intègre également des règles de gestion, comme indiqué par les relations d'extension (extends) et d'inclusion (include) entre les différents cas d'utilisation. Par exemple :

- "Receive notifications for each change of step" et "Receive notifications in case of potential delays" sont des extensions du cas "Track the progress of the order" indiquant que ces actions ne peuvent être entreprises que dans le contexte de la réception de notification
- "Validate the Order" est inclus dans "Log in" ce qui signifie qu'elle est une étape nécessaire au processus de commande.

Diagramme UML Classes :

- User :
 - . Représente un utilisateur du système.
 - . Attributs : Id, Firstname, Lastname, Email, Password
 - . Méthodes : Login(), CreateCommand()
 - . Relation :
- Command :
 - . Représente une commande effectuée.
 - . Attributs : Id, IsValid, PlaceDelivery, Status

- . Méthodes : GetCommand(),
GetPlaceDelivery(),
UpdateStatus(Notification),
. Relation :

- Notifier :

- . Représente les notifications qui peuvent être envoyées à l'utilisateur.

- . Attributs : Notification
 - . Méthodes : Send(Notification)
 - . Relation :

- Factory :

- . Classe mère du Design Pattern Factory, "l'usine" de création des voitures.

- . Attributs : Id, Name, Engine, Color
 - . Méthodes : CreateCar(),
GetColor(Command), GetMotor(Command)
 - . Relation :

- CybertruckCreator :

- . Classe qui appartient au Design Pattern Factory. Créateur d'objet Cybertruck.

- . Attributs : Id, Name, Engine, Color
 - . Méthodes : CreateCar(),
GetColor(Command), GetMotor(Command)

. Relation :

- ModelSCreator :

. Classe qui appartient au Design Pattern Factory. Créateur d'objet Model S.

. Attributs : Id, Name, Engine, Color

. Méthodes : CreateCar(),
GetColor(Command), GetMotor(Command)

. Relation :

- ModelXCreator :

. Classe qui appartient au Design Pattern Factory. Créateur d'objet Model X.

. Attributs : Id, Name, Engine, Color

. Méthodes : CreateCar(),
GetColor(Command), GetMotor(Command)

. Relation :

- Car :

. Interface qui appartient au Design Pattern Factory. Créateur d'objet Car.

.

- Cybertruck :

. Classe qui appartient au Design Pattern Factory. Elle regroupe les objets Cybertruck.

- ModelS :
 - . Classe qui appartient au Design Pattern Factory. Elle regroupe les objets Model S.
- ModelX :
 - . Classe qui appartient au Design Pattern Factory. Elle regroupe les objets Model X.

Diagramme UML Séquence :

Diagramme Commande :

- **Acteurs** : Visitor, Client, Website, Database
- **Processus de commande** : Commence par le choix de la voiture parmi celles proposées.
- **Sélection et informations de commande** : L'utilisateur sélectionne les informations désirées.
- **Optionnel (OPT)** : Si l'utilisateur n'est pas connecté, il passe par le processus de connexion .
- **Références (REF)** : Le processus de connexion/inscription séparé est référencé ici.
- **Vérification de la commande** : Les données envoyés sont vérifiées par le site.

- **Alternatives (ALT)** : Si la commande veut être acceptée, les données doivent être correctes. Sinon, l'utilisateur est redirigé vers le formulaire de commande.
- **Boucles (LOOP)** : Tant que les données sont incorrectes, l'utilisateur est invité à les corriger jusqu'à ce qu'elles soient validées.
- **Sauvegarde de la commande** : Après la confirmation des données de commande, elle est enregistrée en base de données et est visible sur le profil de l'utilisateur.

Diagramme Inscription/Connexion :

- **Connexion** : L'utilisateur initie la connexion et entre les informations requises dans une boucle jusqu'à succès. Les informations sont vérifiées par la base de données. En cas d'échec, un message d'erreur s'affiche; en cas de succès, l'utilisateur est redirigé vers l'accueil.
- **Inscription** : L'utilisateur remplit un formulaire d'inscription.
La base de données vérifie l'unicité des données soumises.
Si les données ne sont pas uniques, l'utilisateur est invité à modifier le formulaire; si elles le sont, le compte est créé.

