

# TRAVAIL PRATIQUE #2

Cours 420-3D4-MO – Interface utilisateur

## CONSIGNES

- **Date de remise du travail : Lundi 19 décembre 2022 avant 18h00**
- Remettre le travail sur Moodle dans un fichier compressé (ZIP) nommé **TP2votreNom.ZIP** qui comprend tout le dossier du projet, l'arbre événementiel et le rapport de débogage (voir grille d'auto-évaluation).

## NIVEAU

Niveau	Événements	Validations dans l'interface	Prototype d'interface
Niveau 3	À déterminer	À déterminer	À déterminer

## UTILISATION DE GITHUB CLASSROOM:

Connectez-vous dans votre compte GitHub. Ensuite, acceptez l'assignation en cliquant sur le lien suivant:

Assignation	Description	Lien
Travail pratique 2	Dépôt pour le travail pratique 2	<a href="https://classroom.github.com/a/nHOd3XfO">https://classroom.github.com/a/nHOd3XfO</a>

## COMMITTS DANS GITHUB CLASSROOM:

Vous devez faire des commits pour les étapes suivantes de votre projet. Vous pouvez faire plus d'un commit pour chacune des étapes:

Étapes	Description de votre commit	
Étape 1	Étape 1 : création du projet vide	
Étape 2	Étape 2 : création de l'interface	
Étape [3...n-1]	Étape [3...n-1] : interaction <contrôle> <nom de l'interaction>	On se base sur les événements sur les contrôles. Voir votre arbre événementiel.
Étape n	Étape n : fin du travail pratique 2	

## RAPPORT DE DÉBOGAGE

Lors de la programmation de votre travail, vous allez rencontrer des erreurs de programme. Je vous demande de faire un rapport de débogage pour **2 erreurs rencontrées**. Pour chaque bogue, vous devez faire une **démonstration vidéo ou photo** du bogue et me montrer la correction appliquée. Dans le rapport, vous devez inclure les éléments suivants :

Rapport de débogage
Erreur rencontrée et type d'erreur (plantage, logique, exécution) : démonstration par capsule vidéo (fichier vidéo ou lien via YouTube ou OneDrive) ou par images (étape qui amène le plantage, avant le bogue et après le bogue)
Méthode pour trouver l'erreur (trace, console, débogueur ou autre)
Correctif apporté : démonstration par capsule vidéo (fichier vidéo ou lien via YouTube ou OneDrive) ou images (corrections apportées après le bogue sous forme d'étapes)

Pour la capsule vidéo, vous pouvez utiliser un des logiciels suivants : Screencast-O-Matic, Filmora ou autre.

## VALIDATIONS POUR LE PROFESSEUR (OBLIGATOIRE):

Vous aurez plusieurs validations pour votre travail pratique. Ces validations sont obligatoires pour la continuation du travail. Les validations peuvent se faire par Teams, en classe ou sur rendez-vous.

### PROTOTYPE D'ÉCRANS

Trois écrans seront demandés. Pour l'écran d'accueil, vous devez obligatoirement respecter ce qui est demandé par l'organisme. Pour les deux autres écrans (Réservations et Affichage des réservations), vous devez faire un prototype d'écran, pour chaque fenêtre, **AVANT de programmer** votre application. Vous devez aussi respecter les spécifications demandées. Les deux prototypes d'écrans devront être validés par le professeur.

### ARBRES DE PROGRAMMATION/ÉVÉNEMENTIELS ET VALIDATIONS/EXCEPTIONS:

Vous devez créer les arbres de programmation/événementiels en utilisant Coggle ou Mindmeister ou autre logiciel

- a. Pour chaque écran, vous devez avoir un arbre événementiel
- b. Pour chaque interaction à compléter, écrivez le nom du contrôle et l'événement de Visual Studio
  - i. Par exemple : `button_afficher_click`
- c. Ensuite, ajoutez les validations possibles ou les exceptions possibles qu'il y a dans votre interaction et ajoutez une description sur ce que vous devez faire pour cette validation (algorithme textuel).
- d. Les arbres événementiels doivent être validés par le professeur.
- e. **Remettre vos arbres événementiels lors de la remise (partage par lien ou en format PDF ou JPG)**

Programmez les interactions selon l'arbre événementiel que vous avez complété.

- a. Après chaque interaction, faites une sauvegarde (commit) de votre projet avec le nom suivant :
  - i. Étape [3...n] : interaction <contrôle> <nom de l'interaction>

### FONCTIONNALITÉ

Vous devez me présenter une fonctionnalité de votre travail pratique pour validation. Le professeur validera certaines parties du code avec vous.

<b>Fonctionnalité à valider:</b>	Faire une réservation pour un camping en particulier avec diverses informations (nombre de nuits, choix du terrain, nombres de personnes, infos du client, calcul du coût total)
	Faire une réservation pour un camping en particulier avec des données erronées. Validation des éléments.

### CALENDRIER MISE EN ŒUVRE

Date / semaine	Calendrier
9 novembre (semaine 11)	Présentation de l'énoncé
Avant le 25 novembre 18h	Validation des prototypes d'écran (Réservation et Affichage des réservations)
Avant le 2 décembre 18h	Validation des arbres événementiels
Avant le 16 décembre 18h	Validation d'une fonctionnalité
19 décembre avant 18h	Remise du travail

## MISE EN SITUATION :

L'organisme Camping Québec vous demande de créer un système de réservation de ses terrains de camping pour ses clients à travers le Québec. Vous devez écrire un programme en C# multi-formulaire qui permet, de sélectionner votre camping, sélectionner votre terrain, faire la réservation pour le camping et afficher les réservations pour chaque camping.

Il y a 4 campings possibles :

Camping	Nombre de terrains	Fichier image
Parc du Bic	10	campingBic.jpg
Parc du Mont-Orford	8	campingOrford.jpg
Camping du Rocher Percé	5	campingPerce.jpg
Camping de la plage de St-Siméon	10	campingStSimeon.jpg

## L'ÉCRAN D'ACCUEIL (OBLIGATOIRE)

L'écran d'accueil doit être **obligatoirement** avoir l'aspect suivant avec les éléments suivants :

- Liste déroulante (**ComboBox**) avec les choix de camping où c'est possible de choisir le camping voulu.
  - Lorsque l'on choisit un camping, l'image (en haut) de celui-ci doit changer pour le camping voulu. L'image doit être
- deux champs **DateTimePicker** qui permet pour la **date d'arrivée** et la **date de départ**.
  - La **date d'arrivée** doit toujours être plus petite que la **date de départ**.
  - Le choix d'une date doit être entre le 1<sup>er</sup> mai et le 31 octobre de la même année.
  - On utilise les **ErrorProvider** pour valider les dates.
- Le bouton « **Réservation** » permet d'afficher la fenêtre de réservation pour le camping voulu.
  - Pour afficher la fenêtre de réservation, on s'assure d'avoir choisi un camping auparavant.
- Le bouton « **Quitter** » permet de quitter l'application.

## L'ÉCRAN DE RÉSERVATION (PROTOTYPE À VALIDER)

Vous devez créer une interface qui permet de finaliser une réservation du camping sélectionné dans l'écran d'accueil.

Vous devez créer l'écran de réservation selon les spécifications demandées ci-dessous :

- **Le design de l'écran et la disposition des contrôles sont à votre choix.**
- Vous devez afficher le nom et l'image du camping choisi dans l'écran d'accueil.
- Vous devez ajouter les terrains disponibles ou réservés visuellement selon les dates choisies auparavant.
- Vous pouvez utiliser des étiquettes (label) avec le numéro des terrains.
  - Mettre une couleur d'arrière-plan différente pour un terrain réservé (par exemple : rouge) et pour un terrain disponible (par exemple : vert).
- La réservation se fait en 4 étapes
  - Sélection du terrain
  - Nombre de personnes (Adultes, enfants de 0 à 17 ans)
  - Informations de la personne qui fait la réservation (nom, courriel, type de paiement)
- Tous les champs doivent être **obligatoires**.
  - On doit utiliser les **ErrorProvider** pour valider tous les champs.
  - Si le champ est valide, on efface le ErrorProvider.
- **Sélection du terrain :**
  - L'utilisateur doit sélectionner son terrain de camping parmi les terrains disponibles selon les dates choisies auparavant (par exemple : une liste déroulante)
- **Nombre de personnes :**
  - Il doit avoir deux champs (NumericUpDown ou textbox) pour le nombre de personnes :
    - Adultes
    - Enfants (0 à 17 ans)
  - La capacité pour une réservation est de 8 personnes maximum. Assurez-vous que le total des 2 champs ne dépasse pas 8.
- **Informations de la personne qui fait la réservation**
  - Champ Nom : Assurez-vous que le champ permette les caractères suivants : lettres, espaces, apostrophe, tiret et caractères de contrôle.
  - Champ Courriel : Assurez-vous que le champ contienne le caractère @.
  - Type de paiement : il doit être dans une liste (boîte de liste ou déroulante).
    - Un choix possible parmi les suivants : Interac, Crédit-Visa, Crédit-MasterCard
- Champ **FACTURE** :
  - C'est un champ d'affichage en **lecture seulement**. On y affiche le nombre de personnes (Adultes et enfants entre 0 à 17 ans), le total de personnes, le nombre de nuits et le coût total.
- Champ Bouton « Faire la réservation »
  - Le bouton permet de finaliser la réservation et d'enregistrer celle-ci dans un fichier texte.
  - Il doit être inactif au début. Dès que la réservation est validée, il devient actif.

## COMMENT CALCULER LE COÛT TOTAL:

Pour calculer le coût total, on doit calculer le coût par nuit pour chaque personne en fonction du camping choisi.

## Prix par nuit par personne

Camping (par nuit)	ADULTE	ENFANT (0 à 17 ans)
Parc du Bic	20\$	5\$
Parc du Mont-Orford	15\$	5\$
Camping du Rocher Percé	30\$	10\$
Camping de la plage Saint-Siméon	25\$	Gratuit

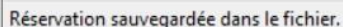
Exemple pour le camping du parc du Bic pour le 5-6-7 mai 2022:

INFORMATIONS	EXEMPLE du champ Facture
-Nombre de personnes ADULTES: xx; ENFANTS (0 À 17 ANS): xx;	ADULTES : 1 ENFANTS (0 À 17 ANS) : 2
-Total de personnes: xx ( <b>calcul du nombre de personnes total</b> )	Total de personnes : 3
-Le nombre de nuits : Date de départ – Date d'arrivée	Nombre de nuits : 2
-Coût total : nombre de nuits X (total du tarif du nombre de personnes)	Coût total : 60\$

## FAIRE LA RÉSERVATION

On sauvegarde dans un des quatre fichiers disponibles (un fichier de réservation par camping). La sauvegarde peut se faire dans un fichier txt, json ou XML.

- **Si fichier texte**, on sauvegarde les informations suivantes sur une ligne (chaque champ doit être séparé d'un point-virgule):
  - NoReservation;NoCampingDateArrivée\_réservation;DateDépart\_réservation;Terrain\_Choisi;NB\_personnes\_adulte;NB\_personnes\_enfant\_0\_17ans;Nom\_client;Courriel\_client;TypePaiement;CoûtTotal
- **Noms obligatoires des fichiers:**
  - RESERV\_BIC
  - RESERV\_ORFORD
  - RESERV\_PERCE
  - RESERV\_SIMEON
- Lorsque la réservation est sauvegardée dans le fichier, vous devez afficher un message dans la **barre de statut** qui dit que la réservation a été faite.

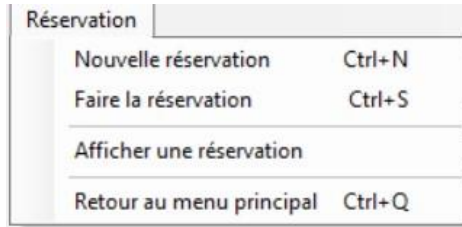


## CONTRAINTES DE L'INTERFACE:

- La réservation doit rester afficher après avoir appuyé sur le bouton **Faire la réservation**.
- Une réservation à 0 nuit ou à 0 nombre de personnes n'existe pas dans notre système.
- On affiche les informations de la réservation dans le champ **FACTURE** en temps réel (lors de la saisie du formulaire) ou lorsqu'on appuie sur le bouton « Faire la réservation ».
- Lorsque vous utilisez une boîte de dialogue pour sauvegarder un fichier de réservation (option « Faire la réservation »), assurez-vous d'être dans le dossier attitré aux réservations.
- Vous devez obligatoirement créer et utiliser les classes Camping et Réservation définies à la fin du document.

## MENU

Voici les menus que vous devez avoir pour l'application (Réservation):



Dans le menu **Réservation**, vous devez avoir les options suivantes :

- **Nouvelle réservation** : Efface tous les champs du formulaire et crée une nouvelle réservation au camping selon les dates demandées.
  - On remet **TOUS les champs comme l'écran de départ**. Donc, les champs doivent être soit vides, à zéro ou actifs/inactif.
  - **On garde les terrains disponibles possibles.**
- **Faire la réservation** : Approuver la réservation et enregistre celle-ci dans un fichier de réservation possible.
- **Afficher une réservation** : l'option permet d'ouvrir un nouveau formulaire qui permet d'afficher toutes les réservations déjà faites pour le camping sélectionné.
- **Retour en menu principal** : retourne au menu principal

## L'ÉCRAN AFFICHAGE DES RÉSERVATIONS (PROTOTYPE À VALIDER)

Vous devez créer une interface qui permet d'afficher les réservations du camping sélectionné dans l'écran d'accueil.

Vous devez créer l'écran de réservation selon les spécifications demandées ci-dessous :

- **Le design de l'écran et la disposition des contrôles sont à votre choix.**
- Vous devez afficher le nom et l'image du camping choisi dans l'écran d'accueil.
- On doit permettre la sélection d'une réservation.
  - Vous pouvez utiliser une boîte de liste, une liste déroulante ou un autre contrôle.
- Lorsque l'on sélectionne une réservation, on doit afficher les informations en lecture seulement dans une zone de texte ou dans des champs texte.
  - **Informations à afficher pour une réservation :**
    - Terrain choisi
    - Nombre de personnes ADULTES, ENFANTS (0 À 17 ANS)
    - Nombre de nuits
    - Type de paiement
    - Coût total

## GRILLE D'ÉVALUATION - TRAVAIL PRATIQUE #2

Écran d'accueil	Note
(5) Interface avec les contrôles demandés (5) Programmation des événements et validations des contrôles à l'écran (5) Exécution de la fenêtre	15
Écran Réservation	
(10) Interface avec les contrôles demandés (10) Programmation des événements et validations des contrôles à l'écran (5) Affichages des données à l'écran (5) Contraintes de l'interface (5) Exécution de la fenêtre	35
Écran Affichage des réservations	
(5) Interface avec les contrôles demandés (5) Programmation des événements et validation des contrôles (5) Affichages des données à l'écran (5) Exécution de la fenêtre	20
Validations auprès du professeur	
(5) Prototypes d'écran : Réservation (5) Prototypes d'écran : Affichage des réservations (5) Arbre événementiel avec validations/exception (5) Exécution de la fonctionnalité demandée	20
Consignes	
(5) Rapport de débogage (5) Respect des consignes (nomenclature, étapes Git, remise, classes)	10
Note finale	100

## GRILLE D'AUTO-ÉVALUATION

<b>Questions à se poser avant la remise</b>	
Est-ce que j'ai accepté l'assignation dans GitHub Classroom?	
Est-ce que j'ai fait le commit pour la création du projet vide?	
Est-ce que j'ai créé mon prototype pour mes deux fenêtres ?	
Est-ce que j'ai fait valider mes prototypes d'images auprès du professeur ?	
Est-ce que j'ai créé l'interface avec les contrôles selon mon prototype?	
Est-ce que j'ai fait le commit pour la création des interfaces?	
Est-ce que j'ai créé mon arbre événementiel selon les contrôles et événements possibles dans les interfaces?	
Est-ce que j'ai fait valider mon arbre événementiel auprès du professeur ?	
Est-ce que j'ai programmé tous les événements possibles selon l'arbre événementiel créé et les interfaces ?	
Est-ce que j'ai ajouté mes validations/exceptions possibles selon les interactions dans mon arbre événementiel?	
Est-ce que j'ai programmé toutes les validations sur les contrôles?	
Est-ce que j'ai fait les étapes Git : commits selon les interactions/actions de l'arbre événementiel?	
Est-ce que j'ai éliminé les bogues d'exécution dans le programme?	
Est-ce que j'ai répertorié les bogues dans mon rapport de débogage ?	
Est-ce que j'ai corrigé les bogues de mon rapport de débogage ?	
Est-ce que j'ai fait la démonstration (bogue et correction) par capsule vidéo ou photos de mes bogues de mon rapport de débogage ?	
Est-ce que j'ai testé les interactions programmées conformes aux interfaces créées?	
Est-ce que j'ai fait le commit pour signifier la fin du projet?	
Est-ce que j'ai remis mon rapport de débogage avec les vidéos (lien ou fichier) ?	
Est-ce que j'ai remis tout le dossier du projet C# dans un fichier compressé?	
Est-ce que j'ai remis l'arbre événementiel dans mon dossier de projet ou par un lien partagé?	
Est-ce que j'ai fait valider une fonctionnalité auprès du professeur ?	
Est-ce que j'ai remis le fichier ZIP sur Moodle?	



## CLASSES OBLIGATOIRES

Vous devez avoir au moins les objets suivants (classes d'objets) dans votre programme. **D'autres champs peuvent être ajoutés si c'est le cas:**

### CAMPING

- Numéro du camping : entier
- Nom du camping : string
- Nombre de terrains : entier
- Chemin de l'image : string

### RÉSERVATION :

- Numéro de réservation : entier (créer automatiquement lors de la réservation)
- Numéro du camping: entier
- Nom du client : string
- Adresse courriel du client : string (doit contenir obligatoirement un @)
- Type de paiement du client : string
- Date de début de la réservation : date
- Date de fin de la réservation : date
- Hébergement choisi : string
- Coût de l'hébergement: nombre réel
- Nombre d'adultes pour la réservation : entier
- Nombre d'enfants entre 0 à 17 ans pour la réservation : entier
- Coût total : nombre réel

À vous de choisir comment vous gérez vos réservations pour chaque camping (dans une liste, dans un tableau, etc.).

---