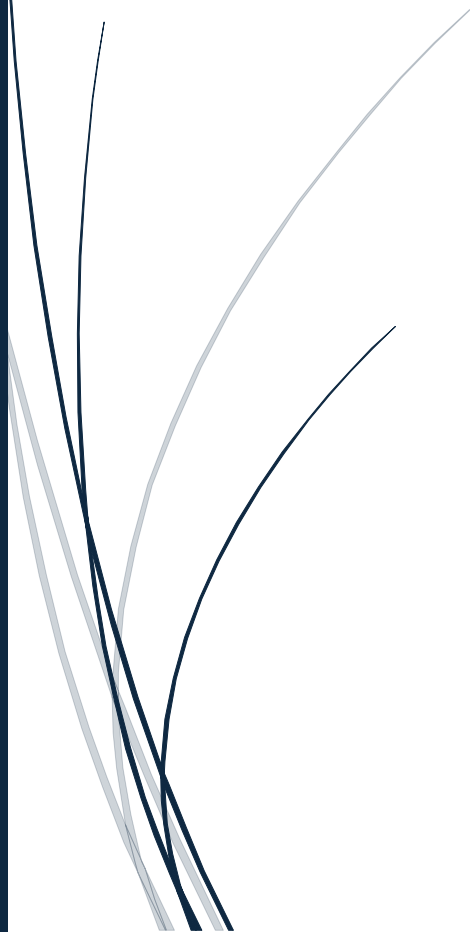
A dark blue vertical bar on the left side of the page. A horizontal arrow, also in dark blue, points to the right from the bar, containing the date.

18/11/2025

Mise à jour automatique d'applications

Several thin, curved lines in dark blue and light grey originate from the bottom left and sweep upwards and to the right.

Boualem El Guendouz, Simon Hanot, Ayoub
Khloufi
CEGEP DE CHICOUTIMI

Table des matières

Intégration d'AutoUpdater.NET dans GameOn	2
Procédure de configuration et d'utilisation	2
Hébergement des fichiers de mise à jour	2
Installation de l'application	2
Exécution de la mise à jour automatique	2
Mise à jour et procédure d'utilisation	3
Remarques finales	3

Intégration d'AutoUpdater.NET dans GameOn

Pour permettre la mise à jour automatique de notre application WPF **GameOn**, nous avons intégré la bibliothèque **AutoUpdater.NET** via NuGet. Cette bibliothèque nous permet de gérer la vérification et le téléchargement des nouvelles versions directement depuis l'application, sans intervention manuelle de l'utilisateur. Nous avons utilisé la méthode `AutoUpdater.Start()` directement dans `MainWindow.xaml.cs`, appelée dès le lancement de l'application. Cette méthode compare la version installée avec la version la plus récente disponible, telle que définie dans le fichier `update.xml`, et affiche automatiquement une fenêtre de mise à jour lorsqu'une nouvelle version est détectée.

Nous avons configuré **AutoUpdater.NET** pour pointer vers un fichier `update.xml` hébergé sur GitHub. Ce fichier contient toutes les informations relatives à la dernière version disponible, ainsi que le lien vers le `setup.exe` correspondant. La version de l'application est définie dans le fichier `.csproj` à l'aide des balises `AssemblyVersion`, `FileVersion` et `Version`, ce qui permet à **AutoUpdater.NET** de comparer correctement les versions et de gérer la mise à jour de manière fiable.

Procédure de configuration et d'utilisation

Hébergement des fichiers de mise à jour

Pour que la mise à jour automatique fonctionne, nous avons hébergé les fichiers nécessaires sur GitHub. Nous avons créé un dossier `update` dans notre dépôt et y avons placé le fichier `update.xml` ainsi que le `setup.exe` généré avec Inno Setup. Le fichier `update.xml` contient la balise `<version>` indiquant la version la plus récente et la balise `<url>` pointant vers le fichier RAW du `setup.exe` sur GitHub. La balise `<mandatory>` permet de définir si la mise à jour doit être obligatoire ou non.

Installation de l'application

Nous avons compilé le projet WPF et généré le `setup.exe` avec Inno Setup afin de permettre l'installation de l'application **GameOn** sur le poste utilisateur. L'utilisateur peut alors installer l'application normalement en suivant les étapes de l'installateur.

Exécution de la mise à jour automatique

Lors du lancement de **GameOn**, la méthode `StartAutoUpdater()` est appelée et l'application consulte automatiquement le fichier `update.xml`. Si une version plus

récente est disponible, une fenêtre de mise à jour s’affiche, permettant à l’utilisateur de télécharger et d’installer la nouvelle version en un seul clic.

Si l’utilisateur avait précédemment choisi de “passer cette version”, il est nécessaire de supprimer la clé `SkippedVersion` dans le registre Windows à l’emplacement suivant pour que la fenêtre de mise à jour réapparaisse : `HKEY_CURRENT_USER\Software\GameOn`

Mise à jour et procédure d’utilisation

Pour modifier la version de l’application et gérer les mises à jour, nous procédons comme suit : tout d’abord, nous mettons à jour la version de GameOn dans le fichier `.csproj` en modifiant les balises `AssemblyVersion`, `FileVersion` et `Version` pour correspondre à la nouvelle version souhaitée. Ensuite, nous mettons à jour le fichier `update.xml` hébergé sur GitHub en indiquant cette nouvelle version dans la balise `<version>` et en mettant à jour l’URL du `setup.exe` correspondant dans la balise `<url>`.

Une fois ces modifications effectuées, il suffit de compiler le projet et de générer un nouveau `setup.exe` avec Inno Setup, que l’on place dans le même dossier `update` sur GitHub. Lorsqu’un utilisateur lance GameOn, l’application consulte automatiquement le fichier `update.xml`. Si la version indiquée dans le fichier est supérieure à celle installée, `AutoUpdater.NET` affiche une fenêtre de mise à jour, permettant à l’utilisateur de télécharger et d’installer la nouvelle version en un seul clic

Remarques finales

Nous avons veillé à ce que chaque nouvelle version soit correctement incrémentée dans le fichier `.csproj` et dans `update.xml`. Les fichiers de mise à jour peuvent être hébergés sur GitHub. Cette intégration permet à GameOn de se mettre à jour automatiquement de manière sécurisée, offrant une expérience utilisateur fluide et professionnelle.