

Développement pour Windows Phone 7

1 Introduction

Ce travail dirigé vous présente comment développer des applications Windows Phone avec Silverlight. En le réalisant, vous vous familiariserez avec les outils nécessaires à la création et aux tests d'application Windows Phone. Vous allez créer une nouvelle application Silverlight pour Windows Phone avec Microsoft Visual Studio 2010 Express pour Windows Phone, puis utiliser cet outil gratuit pour définir la logique applicative et l'interface utilisateur. Vous allez apprendre à déployer une application dans l'émulateur Windows Phone et comment utiliser le débogueur pour positionner des points d'arrêt, faire du pas-à-pas à travers le programme et analyser les valeurs de variables. Enfin, vous allez voir comment ouvrir un projet dans Expression Blend pour modifier le style des contrôles et créer des animations.

2 Installation de "Windows Phone Developer Tools"

Vous trouverez la procédure d'installation à la page :

<http://www.microsoft.com/downloads/en/details.aspx?FamilyID=04704acf-a63a-4f97-952c-8b51b34booce&displaylang=en>

Le Windows Phone Developer Tools contient :

- Visual Studio 2010 Express for Windows Phone
- Windows Phone Emulator Resources
- Silverlight 4 Tools For Visual Studio
- XNA Game Studio 4.0
- Microsoft Expression Blend for Windows Phone,

les outils nécessaires au développement sous Windows Phone 7.

3 Exercice 1 – Créer un projet d'application Windows Phone dans Visual Studio

3.1 Créer votre première application

Vous utilisez ici un modèle prédéfini dans Microsoft Visual Studio 2010 Express pour Windows Phone pour créer un projet d'application Silverlight pour Windows Phone comme base de démarrage de votre première application. Etudiez le contenu des fichiers Apps.xaml, MainPage.xaml, MainPage.xaml.cs. Utilisez entre autre le View Designer et la correspondance du rendu avec le fichier XAML.

3.2 Lancez et testez

A ce stade du TD, l'application ne dispose pas de beaucoup de fonctionnalités, mais suffisamment toutefois pour un premier test d'exécution ! Ainsi dans cette tâche, vous compilez l'application, la déployez dans l'émulateur Windows Phone puis l'exécutez pour comprendre le cycle de développement.

Développement pour Windows Phone 7

4 Exercice 2 – Définir une nouvelle interface utilisateur

4.1 Définir l'interface utilisateur

Vous allez créer ici les éléments d'UI de l'application, très simple, HelloPhone.

Une fois terminée, l'interface utilisation contient un label (textblock), une zone de texte et un bouton. Pour utiliser l'application, vous saisissez du texte dans la zone appropriée puis, lorsque vous cliquez sur le bouton, l'application affiche le texte saisi préalablement. Les plus familiers avec XAML pourront rajouter une ligne dans la grille du « layout root » pour positionner leur nouveaux contrôles.

4.2 Gérer les évènements depuis l'interface utilisateur

Dans cette tâche, vous définissez la gestion des évènements liés aux actions effectuées depuis l'interface utilisateur, c'est-à-dire en l'occurrence le clic sur le bouton. Cette gestion d'événement est définie en utilisant du code dans le fichier de code sous-jacent de la page. Dans les propriétés de type « events » du bouton vous pouvez définir le nom du call back associé au clic. Visualisez la modification qui en résulte sous XAML, ensuite modifiez le code de votre call back dans le fichier MainPage.xaml.cs pour transmettre la chaîne de caractère de la textbox au label.

4.3 Vérification

Dans cette tâche, vous compilez, déployez et exécutez l'application dans l'émulateur Windows Phone afin de le tester son bon fonctionnement. En complément, vous utilisez un point d'arrêt et le débogueur pour naviguer dans le code source (Cf. menu Debug), et examinez la valeur des variables (en positionnant le curseur de la souris sur le code correspondant). Ceci vous donne un bref aperçu de la façon dont vous pouvez utiliser Visual Studio pour déboguer les applications exécutées dans l'émulateur.

5 Exercice 3 – Création d'un bouton personnalisé avec Expression Blend

Nativement, les contrôles Silverlight séparent leur logique de leur apparence en utilisant des modèles. Un ControlTemplate spécifie les structures et comportements visuels d'un contrôle. Vous pouvez personnaliser l'apparence de la plupart des contrôles en modifiant leurs ControlTemplate par défaut. Cela permet de modifier l'apparence d'un contrôle sans en changer les fonctionnalités ; par exemple, il est possible de définir des boutons ronds plutôt que carrés tout en conservant leurs fonctionnalités, telles que l'événement Click.

5.1 Modification de l'apparence d'un bouton

Dans cette tâche, vous ouvrez dans Expression Blend le projet Visual Studio créé dans l'exercice précédent et remplacez le ControlTemplate du bouton pour modifier son apparence. Via la création d'un ControlTemplate en XAML, vous pouvez modifier l'apparence d'un contrôle sans écrire de code.

Pour cela ouvrez MainPage.xaml dans Expression Blend (click droit sur le fichier).

Assurez-vous que MainPage.xaml est bien ouvert dans la fenêtre de design (si ce n'est pas le cas, double-cliquez sur le fichier pour l'ouvrir) et que l'espace de travail courant est en mode Design. Pour cela, vérifiez que l'option Design est cochée dans le sous-menu Workspaces du menu Window.

Dans la fenêtre de design, faites un clic-droit sur votre bouton, allez sur Edit Template et sélectionnez Create Empty.

Développement pour Windows Phone 7

Dans la boîte de dialogue Create ControlTemplate Resource, indiquez le nom FancyButton par exemple, laissez la valeur actuelle de "This document" dans l'option Define in, puis cliquez sur OK.

Modifiez le mode d'affichage du modèle. Pour cela, dans le panneau Objects and Timeline, faites un clic-droit sur l'élément Grid du Template, et sélectionnez Border au sein du sous-menu Change Layout Type.

L'élément Border toujours sélectionné, passez sur ses Properties et dans le menu Appearance. Vous pouvez alors modifier l'apparence de votre bouton comme vous le souhaitez. Modifiez son contour et sa couleur de fonds.

5.2 Ajouter des effets visuels au bouton

Dans cette tâche, vous modifierez le modèle de contrôle pour y ajouter des effets visuels afin d'avoir un aperçu lorsque le bouton a le focus, et de modifier sa position sur la page lorsqu'il est pressé.

Sélectionnez le contrôle Border dans le panneau Objects and Timeline. Passez maintenant au panneau States et regardez les effets visuels disponibles pour ce contrôle. Dans la catégorie FocusStates, sélectionnez l'état Unfocused pour activer l'enregistrement pour l'état engendré lorsque le bouton n'a pas le focus. Tout changement opéré sur le modèle selon ce mode ne s'applique qu'à cet état particulier. Modifiez pour cet état la couleur de fonds du bouton et décalez légèrement sa position.

6 Exercice 8 – Créer une animation pour le texte du label

Les animations sont basées sur des images principales définissant les points de début et fin d'une transition visuelle. Pour créer une animation dans Expression Blend, vous créez un Storyboard, et y définissez des images principales sur une timeline pour indiquer les changements de propriétés. Lorsque vous exécutez un jeu d'animations, Expression Blend gère les changements de propriétés sur la période de temps spécifiée et affiche les résultats dans l'application. Vous pouvez animer toute propriété d'un objet de cette manière, y compris les propriétés non-visuelles.

Nous allons ici gérer une animation sur le texte du label de notre application. Pour cela mettez vous en mode animation. Dans l'onglet Objects and Timeline rajouter une Storyboard (bouton +) sur le label. Jouez alors sur l'évolution des propriétés sur la timeline de la Storyboard. Par exemple, vous pourriez fixer une image à la marque correspondant à la seconde 0 pour enregistrer la position d'un rectangle sur le bord gauche de l'espace de travail, puis fixer une image à la marque de la seconde 1 pour enregistrer la position de ce même rectangle sur le bord droit de cet espace de travail. L'animation générée sera basée sur la transformation des propriétés X et Y du rectangle en 1 seconde.

7 Exercice 9 – Vérification

Lors du design dans Expression Blend, vous avez pré visualisé l'interface utilisateur de votre application Silverlight et exécutée dans un navigateur. Dans cette tâche, vous exécutez l'application dans l'émulateur Windows Phone pour voir le résultat de vos modifications dans l'interface utilisateur.

Développement pour Windows Phone 7

8 Exercice 10 – Bandit Manchot

Lors de ce TD vous avez acquis une partie des notions nécessaires pour développer une application sous Windows Phone 7. Nous vous proposons donc ici de reprendre ces notions dans le développement d'une application de Bandit Manchot.

Nous utiliserons pour cela la classe Random du .Net Framework pour les tirages aléatoires. Notre jeu affichera une image de gains dès lors que trois 7 seront affichés sur un tirage.

Vous utiliserez autant que possible des effets visuels rendant le jeu attractif.