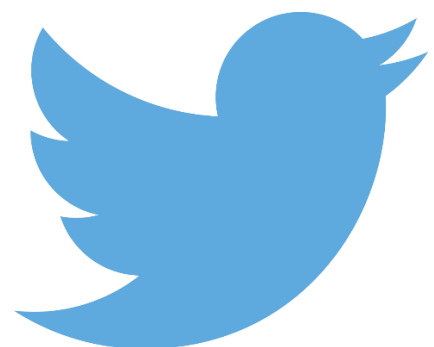


PROJET .NET



INTRODUCTION

Vous êtes engagé par une société qui souhaite créer une application permettant d'interagir avec Twitter. Après étude la société voudrait faire une application Universal Apps (utilisant le langage C#). Vous vous retrouvez donc en charge de la création de cette application avec votre collègue.

PREPARATION



Dans un premier temps, vous allez créer un répertoire [Git](#) qui facilitera votre travail d'équipe. Des « commits » réguliers sont très importants, et permettent d'avoir des checkpoints sur le projet, de vérifier le bon déroulement de celui-ci et la participation de chacun au projet.

DEMANDE DU CLIENT

L'application sera réalisée en Universal Apps (utilisant le langage C#).

L'application devra permettre :

- la connexion à un compte twitter
- l'affichage de la timeline du compte
- l'ajout de tweet à la timeline du compte
- la suppression de tweet à la timeline du compte
- de répondre à un tweet
- de retweeter
- de « liker » un tweet
- de faire une recherche sur twitter à partir d'un mot clé
- d'afficher la timeline d'un autre profil

PHASE D'ANALYSE

Vous devrez commencer par analyser les besoins pour chacune des technologies, comme la consommation de service web (phase de connexion OAuth v2, Api REST, JSON), la création de l'interface et des Frameworks disponibles dans les différents langages.

AIDE

En ce qui concerne la récupération de flux, vous pourrez, dans un premier temps, utiliser un fichier JSON mis à votre disposition. Vous pourrez ainsi construire votre interface graphique, sans vous soucier de la consommation de l'API.

La structure JSON du fichier est exactement la même que celle renvoyée par l'API de Twitter pour une timeline.

Lorsque votre UI sera en place, vous pourrez alors faire appel à l'API de Twitter afin de récupérer un flux plus dynamique.

Pour consommer l'API de Twitter, n'hésitez pas à vous aider de sa [documentation](#) ainsi que de [cette page](#) qui vous expliquera comment utiliser les « timelines ». Si vous avez besoin de faire différent test pour vos requêtes API, [cet outil](#) peut également vous aider.

RENDU DU PROJET

Pour votre application en C#, un dossier contenant la solution VISUAL STUDIO et le(s) projet(s) ainsi que les fichiers « dll » des bibliothèques externes que vous avez peut-être utilisées, ainsi que tous les fichiers dont vous avez besoin pour le bon fonctionnement de votre application. **(À l'exception du dossier bin de chacun des projets)**

ATTENTION A LA TRICHE

N'essayez pas de tricher, **la triche partielle ou totale sera assortie d'une note de 0 éliminatoire.** J'insiste aussi sur le fait que les personnes concernées subiront le même sort qu'elle ait ou non fait le travail à l'origine.

De plus le « copier-coller » d'internet est lui aussi **INTERDIT**. Trouver des informations sur le fonctionnement d'une classe ou d'une méthode est naturellement autorisé, mais faire la copie intégrale d'un tuto ou article sur le net est interdit et sera considéré comme de la triche.

BAREME

Instructions	Points
SoC	2
XAML – Code Behind	6
Pattern (Model, ViewModel, etc)	8
Propreté du code	2
Commentaires	2

FORMAT DU RENDU

Vous enverrez votre solution dans un zip nommé de la manière suivante **AIX_UWP_NOM1_NOM2.zip** en remplaçant *NOM* par votre nom ! Merci aussi de bien vouloir ne pas zipper les dossier **bin** et **obj** de chaque projet, je recompilerai moi-même les solutions.

Vous l'enverrez à l'adresse suivante : alexis.vernot@ynov.com avec pour objet de mail : **[B3][Aix] Partiel – UWP**

Le manque de respect des consignes précédemment citées sera sanctionné sur la note finale.