

Web Media Manager

Jean-Philippe Froelicher

10 juin 2015

Centre de Formation Professionnel Technique

Sommaire

- 1 Introduction
- 2 Principales fonctionnalités
- 3 Mise en oeuvre
- 4 Démonstration
- 5 Conclusion & perspectives
- 6 Questions

Média vidéo du Web

- Plate-forme d'hébergement vidéo.
- Twitch ; Youtube ; Dailymotion ; Vimeo ; Ustream etc.
- Vidéo à la demande (VOD); diffusion de flux vidéo en direct.
- Rémunération suivant le nombres de vues; dons.



Sujet

- Projet de diplôme
- Agrégateur de sites
 - Reproduis les mêmes fonctions
 - Ajout de fonctions
 - Évolutive et souple
- Langage C#

Fonctions de bases

Chaque site intègre ces fonctions :

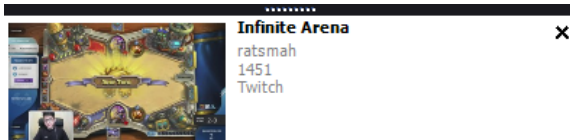
- Connexion à un compte
- Visionnage d'une vidéo / vidéo en direct
- S'abonner à une chaîne
- Chat IRC pour les vidéos en direct
- Recherche de vidéos
- Vidéos actuellement en direct ; dernières vidéos
- Chaînes abonnées

Notifications

Une notification apparaît lorsqu'une vidéo sort ou qu'une nouvelle diffusion en direct commence

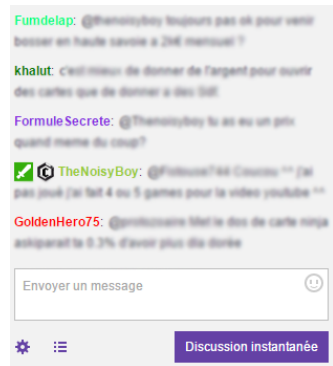
- Pop-up qui apparaît avec les informations de la vidéo.

Les notifications sont activées uniquement pour les chaînes que l'utilisateur suit.



Chat IRC

Internet Relay Chat, Protocole de communication textuelle.
Uniquement sur les vidéos en direct.

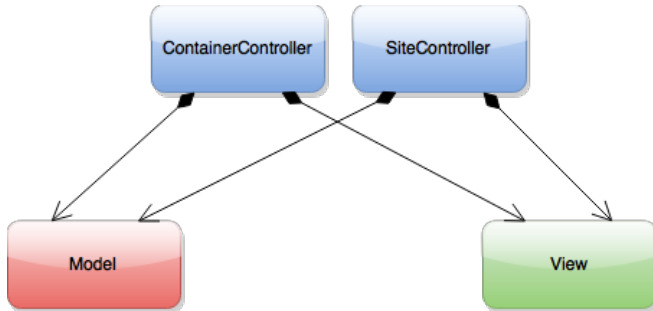


Catégorie & Liste de lecture

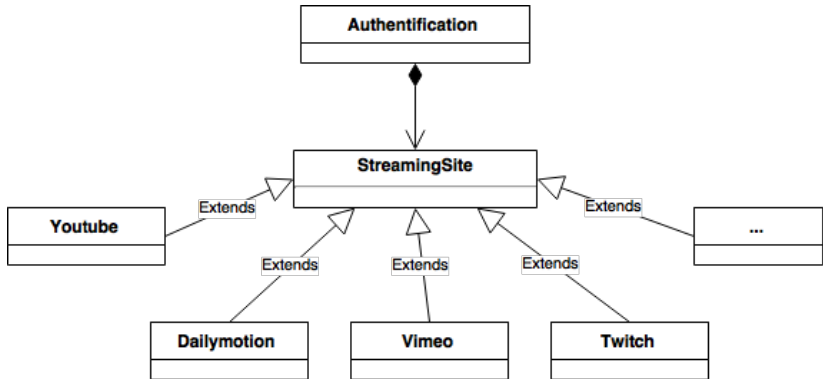
Une catégorie sert à rassembler des vidéos de différents sites au même endroit.

Une liste de lecture lit les vidéos qu'elle contient à la suite.

Conception



Conception

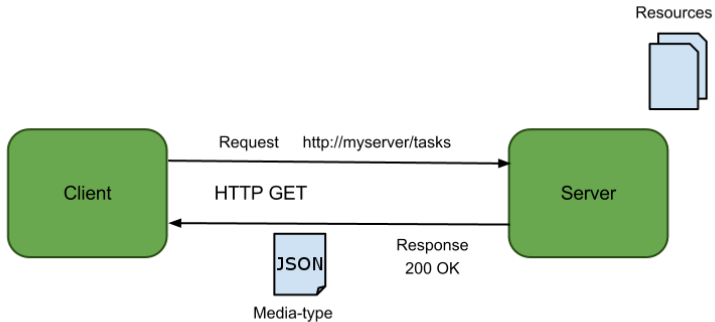


SVideo & SChannel

- Structure
 - Informations communes
- Vidéo et chaîne
- Type générique

```
public struct SVideo
{
    public string id;
    public string videoName;
    public string channelName;
    public int nbViews;
    public string description;
    public string preview;
    public string playerLink;
    public DateTime createdAt;
    public string link;
    public bool live;
    public string siteName;
    public string url_irc;
    public bool channelIsFollowed;
}
```

Récupération des données



Json

```
{
  "type": "user",
  "name": "test_user1",
  "created_at": "2011-06-03T17:49:19Z",
  "updated_at": "2012-06-18T17:19:57Z",
  "_links": {
    "self": "https://api.twitch.tv/kraken/users/test_user1"
  },
  "logo": "http://static-cdn.itvnw.net/1tv_user_pictures/test_user1-profile_image-62e8318af864d6d7-300x300.jpeg",
  "_id": 22761313,
  "display_name": "test_user1",
  "email": "asdf@asdf.com",
  "partnered": true,
  "bio": "test bio woo I'm a test user",
  "notifications": {
    "email": true,
    "push": false
  }
}
```

Exemple de données formaté en Json.

Classes structures

Afin de recevoir les données dans un bon type, il faut créer des classes structures pour chaque site.

```
[DataContract]
public class Channel
{
    [DataMember]
    public string mature { get; set; }

    [DataMember]
    public string status { get; set; }

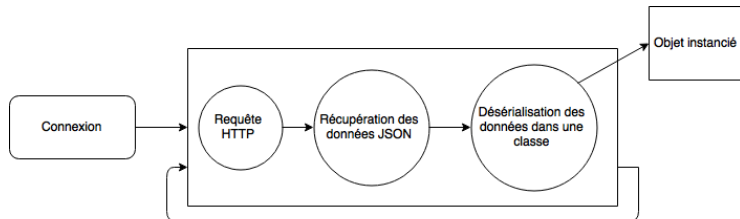
    [DataMember]
    public string broadcaster_language { get; set; }
```

Désérialisation

Un objet est créé depuis les données formaté en Json.
Type générique et lors de l'appel de la méthode, on indique une classe structure.

```
1  /// <summary>
2  /// Deserialize a http web response
3  /// </summary>
4  /// <typeparam name="T">Generic type</typeparam>
5  /// <param name="jsonContent">content json</param>
6  /// <returns>the object deserialized</returns>
7  public static T Deserialize<T>(Stream jsonContent)
8  {
9      var httpResponse = jsonContent;
10
11      //Read the response
12      using (var streamReader = new StreamReader(httpResponse))
13      {
14          //Add to the generics variable the result
15          T answer = ←
16              JsonConvert.DeserializeObject<T>(streamReader.ReadToEnd());
17          return answer;
18      }
19  }
```

Cycle d'un accès



Connexion

- Avoir un compte existant sur le site
- Connexion grâce au protocole OAuth2
- Utilisation du lien de connexion de l'API
 - `api.twitch.tv/kraken/oauth2/authorize?response_type=token`
`&client_id= 9jfbie2pedk3xzoj3s53268v7fb4zds`
`&redirect_uri=`
`https ://froelicher.github.io/WebMediaManager/WebSite`
`&scope=user_read+channel_read`
- Récupération du jeton d'accès dans l'url de redirection
 - Création d'un GitHub Page

Notifications

Vérification toutes les 5 secondes

- Comparaison de deux listes
- Multi-Thread

Composant venant du Web

- NotificationWindow

Catégorie & Liste de lectures

Stockage dans un fichier de type INI dans le dossier appdata de l'utilisateur.

Les vidéos sont stockées sous forme de liens.

[Ping pong]

link=<http://www.twitch.tv/amazhs/v5359902>;<https://www.youtube.com/watch?v=JndFEKht7co>

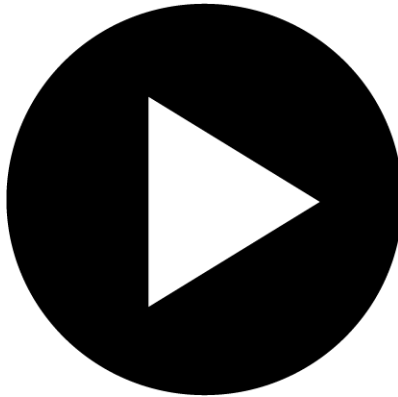
[Tennis]

link=<https://www.youtube.com/watch?v=5XKioMcTh0U>;<http://www.twitch.tv/hearthstonefr/v3984526>

[Football]

link=<http://www.twitch.tv/hearthstonefr/v3958585>;<https://www.youtube.com/watch?v=rCgHhbcP7V4>

Démonstration



Conclusions

- Application fonctionnelle
- Certaines fonctions non implémentée

Perspectives

- Compléter les fonctions manquante
- Implémenter un bon nombre de sites
- Changer de langage

Questions

