Wavo Basic Link Harvester Bookmarklet Requirements Specification

Gabriel Lespérance <gabriel@wavo.me>

February 1st, 2012

1 Purpose

The purpose of this software project is to provide a basis for the development of the *Wavo Link Harvester* that will allow users to submit the music-video *links and embedded objects* contained in a webpage to the wavo servers for further processing.

This module will eventually be extended to allow users to save music-videos from across the web and save them in their wavo collections.

As a secondary purpose, the following module will be given as an assignment to *Baptiste Bonnaudet <baptiste.bonnaudet@gmail.com>* in order to assess his programming skills as well as his ability to work in an autonomous fashion when provided with general requirements.

2 Requirements

2.1 Functional Requirements

2.1.1 Gathering Elements

Gathering Links Elements The following URI types must be gathered :

- youtube links
- Vimeo links
- soundcloud links

Gathering Embedded Objects The following embedded objects types must be gathered :

- youtube links
- Vimeo links
- soundcloud links

Gathering iframe Elements The following iframe types must be gathered :

- youtube links
- Vimeo links
- soundcloud links

2.1.2 Logging Content to the Console

The elements gathered must be printed *console.log* as they are found in the DOM.

This logging facility must be easily extensible in order to allow reporting to the wavo servers in the future.

Non-Functional Requirements

Documentation   Each of the module's routine/functions needs to be properly documented using jsDocs[1]. The required jsDocs elements are *@param*, *@return*, and *@constructor* when applicable.

Extensibility   The module must be easily extensible in order to allow for the gathering of object types other than *links embedded objects.*, or *iframes.*

Each of these sub-modules must also be easily extensible in order to allow, for example, the gathering of additional *types* of link.

Maintainability   The module must be easily maintainable for all users. Hence, the code is supposed to loaded from a central public repository when the user activates the bookmarklet.

The using the following code snippet as the bookmarked URI should allow the fulfillment of this requirement :

```
javascript:(function(){
  var a = document.createElement('script');
  a.setAttribute('src', JS_MODULE_URI);document.body.appendChild(a);}
)();
```

Licensing   In order to avoid costly legal process, this software will have to be released under the MIT License[2]. Hence it should bare the following license on top of all its files :

---

[1]see http://google-styleguide.googlecode.com/svn/trunk/javascriptguide.xml?showone=Comments#Comments

[2]http://en.wikipedia.org/wiki/MIT_License