Implementation of a Digital Container

The application Interface contains a tab control with two tabs:

* Encode Tab;
* Decode Tab.

1. ***Encode Tab*** provides the following functionality:

* Add Single File to the container;
* Add Multiple Files to the container;
* Encode Container;
* Remove a file from the container.

***Add Single File*** to the container is achieved using an OpenFileDialog which prompts the user to select a file. After that, the application reads some info about the file (name, size, and content), and adds the file inside the digital container. The Container is stored in memory.

***Add Multiple Files*** is same as ***Add Single File***, but in this case, the user can select multiple files using the OpenFileDialog.

***Remove File*** from the container is the method used to remove file from the container before you encode it.

***Encode Container*** is used to save the container file. This step is done via SaveFileDialog which prompts the user to select a location for saving the resulting container file. This file has .dc extension.

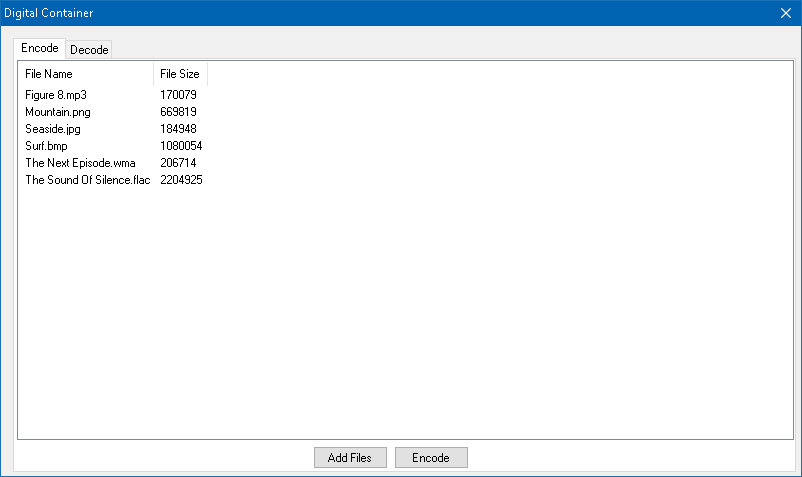
Below is the ***Encode Tab*** screenshot :

Figure 1 - Encode Tab Screenshot

1. ***Decode Tab*** provides the following functionality:

* Open digital container;
* Decode the entire digital container;
* Decode a single file from the digital container.

***Open Digital Container,*** uses a OpenFileDialog which prompts the user to select the previously created container (the file with .dc extension). After opening the container, the application first reads the container header which contains info about files inside.

***Decode The Entire Container,*** prompts user to select a directory for storing decoded files. Next, it will iterate over the all encoded files and decode them.

***Decode Single File,*** user select from the ListView a file for decoding it and application ask user where to save the decoded file.

Below is the ***Decode Tab:***

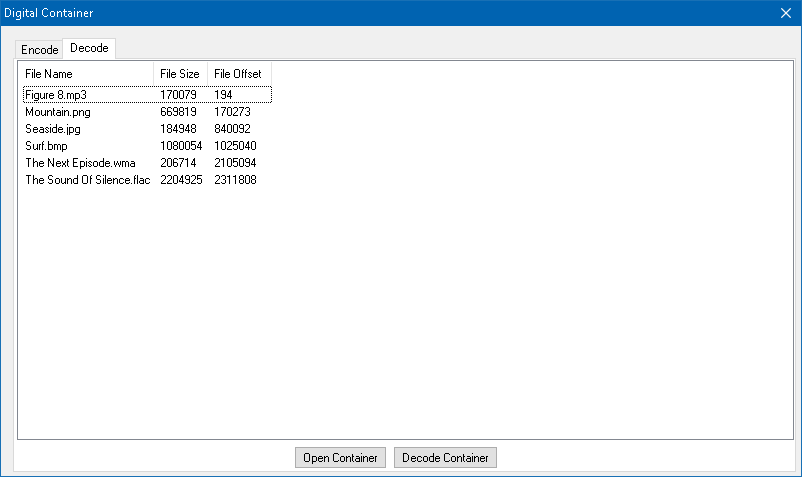


Figure 2 - Decode Tab Screenshot

***Container Header Description:***

|  |  |
| --- | --- |
| ***Container Header:*** | |
| **Field:** | **Data Type:** |
| NumberOfFiles(n) | Int32 |
| FileName(1) | String |
| FileSize(1) | Int64 |
| FileOffset(1) | Int64 |
| ..... | .... |
| FileName(n) | String |
| FileSize(n) | Int64 |
| FileOffset(n) | Int64 |
| ***File Data:*** | |
| **Field:** | **Data Type:** |
| FileData(1) | Byte[] |
| .... | .... |
| FileData(n) | Byte[] |

Table 1 - Digital Container Header Description

***Number Of Files*** is the first field of the header, it shows the number of files encoded inside the container.

After it, the header contains ***Number Of Files,*** FileName, FileSize, FileOffset fields.

***FileName*** stores the original filename.

***FileSize*** stores the length of the encoded file.

***FileOffset*** stores the start position of the encoded file.

***File Data Description:***

Contains ***Number Of Files*** File Data Fields.

***File Data,*** contains the data of the encoded file.

Encoded files are extracted by seeking through the container using the ***FileOffset*** and ***FileSize,*** info from the Container Header. A file is stored between ***FileOffset*** and ***FileSize***.

).

*.*